AI for Collider Physics

Efficient Computation & New Physics Discovery

Myeonghun Park

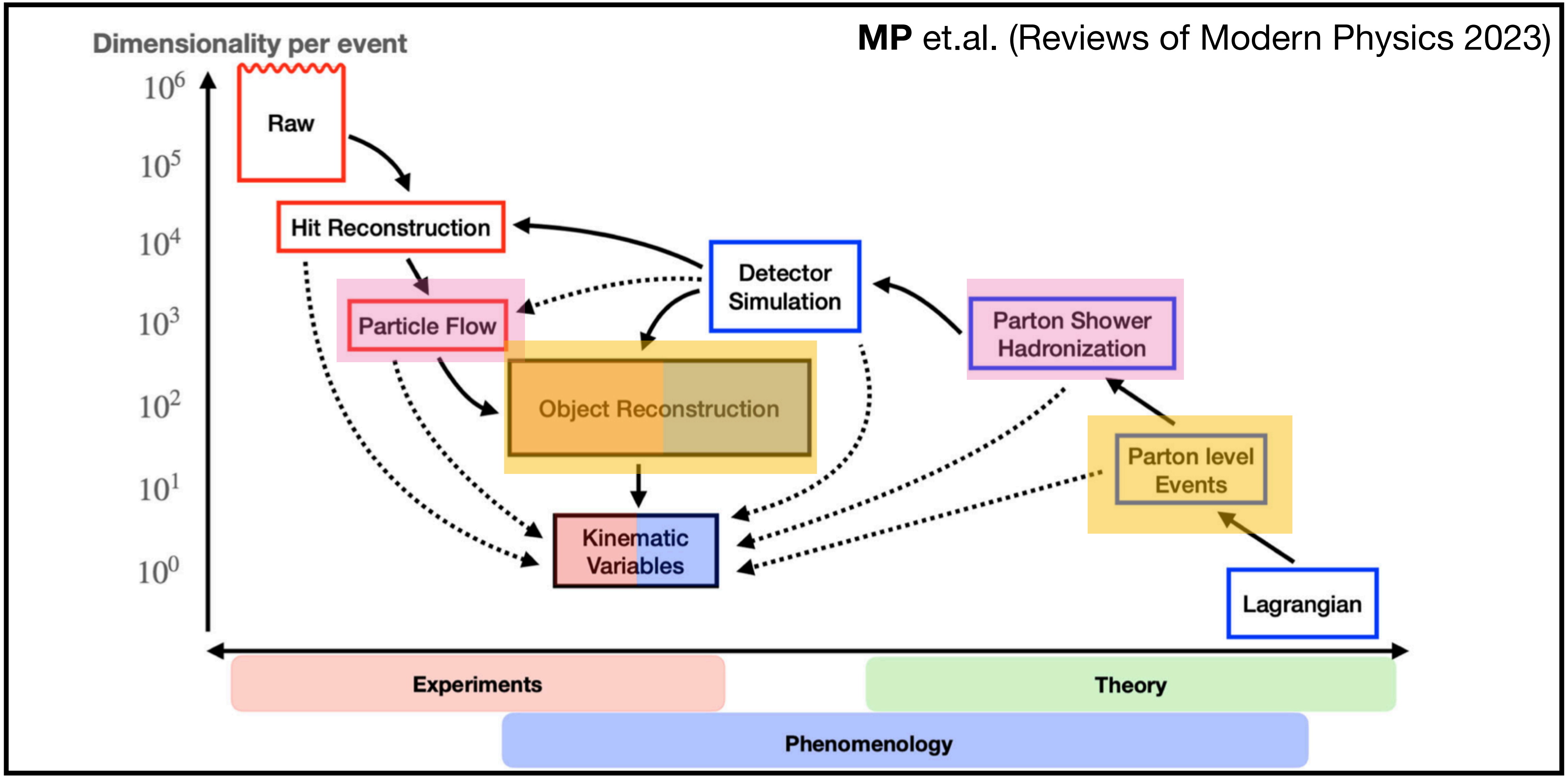
SeoulTech

한국고에너지물리학회 2026 봄 학술대회

2026.06.11 - 13

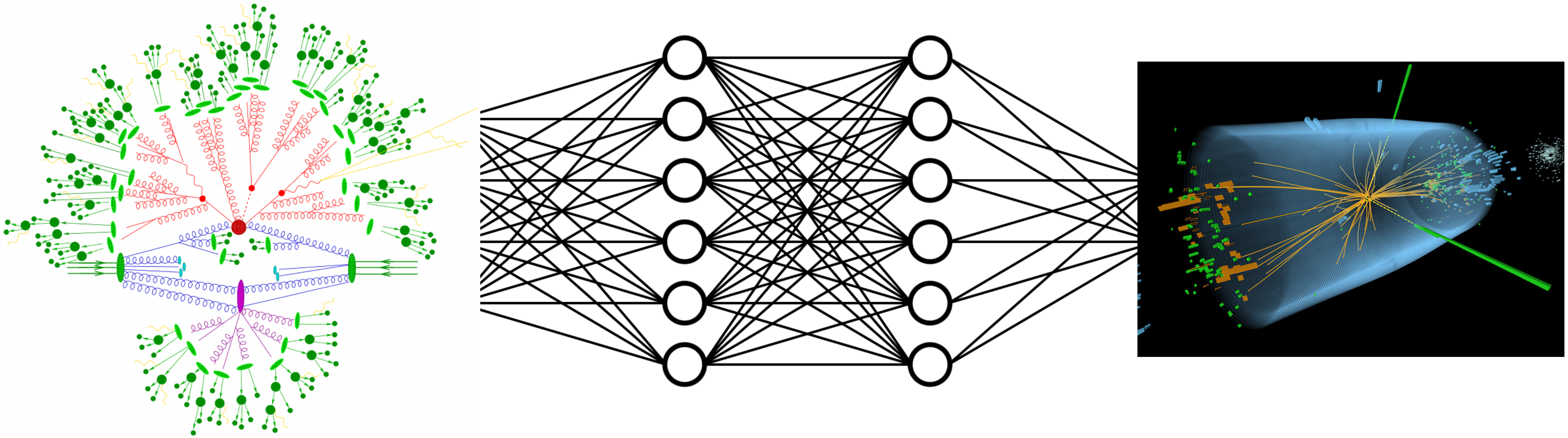
Understanding Physics using the LHC

- Recent advancements in Machine Learning (**ML**) allow us to fully exploit **all available information**, including raw-level data such as detector responses.



Theoretical expectations (Supervised) ML

Theory-Compare-Data



- With our elaborated **theoretical model**,

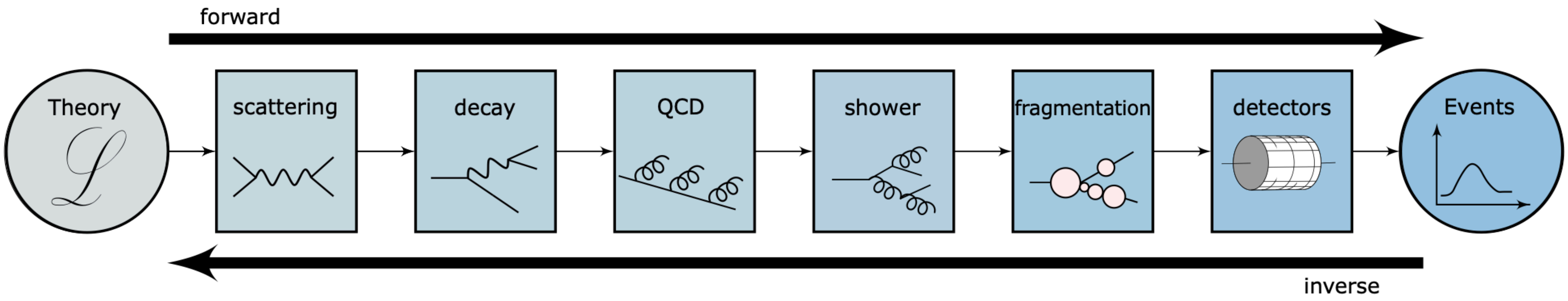
  1) Get **expectations** from **MC simulations**

  2) Get **data** from **experiments** (e.g. the LHC)

  3) Compare our expectation to data with sophisticated computer **algorithms** (including Machine Learning)

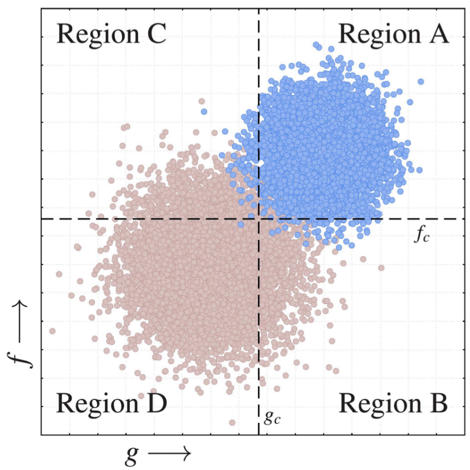
- **Data come from the chain of Monte Carlo simulations**

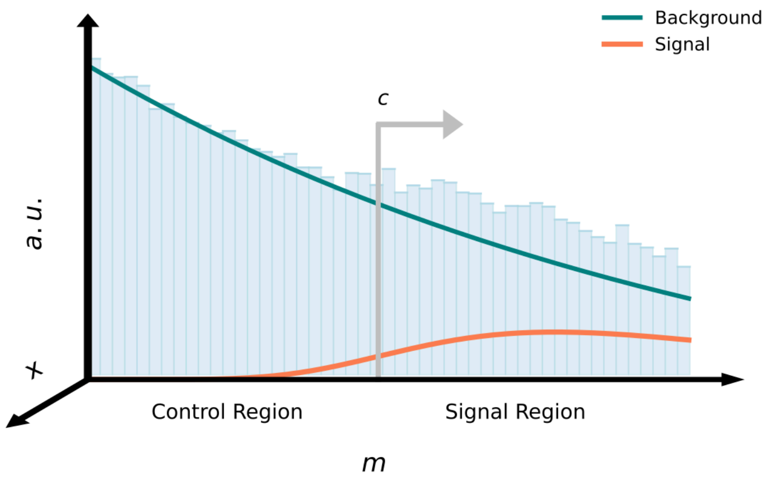
Modern Machine Learning for LHC Physicists (Tilman Plehn et.al. arXiv:2211.01421)



- **Data come from LHC data (controlled samples)**

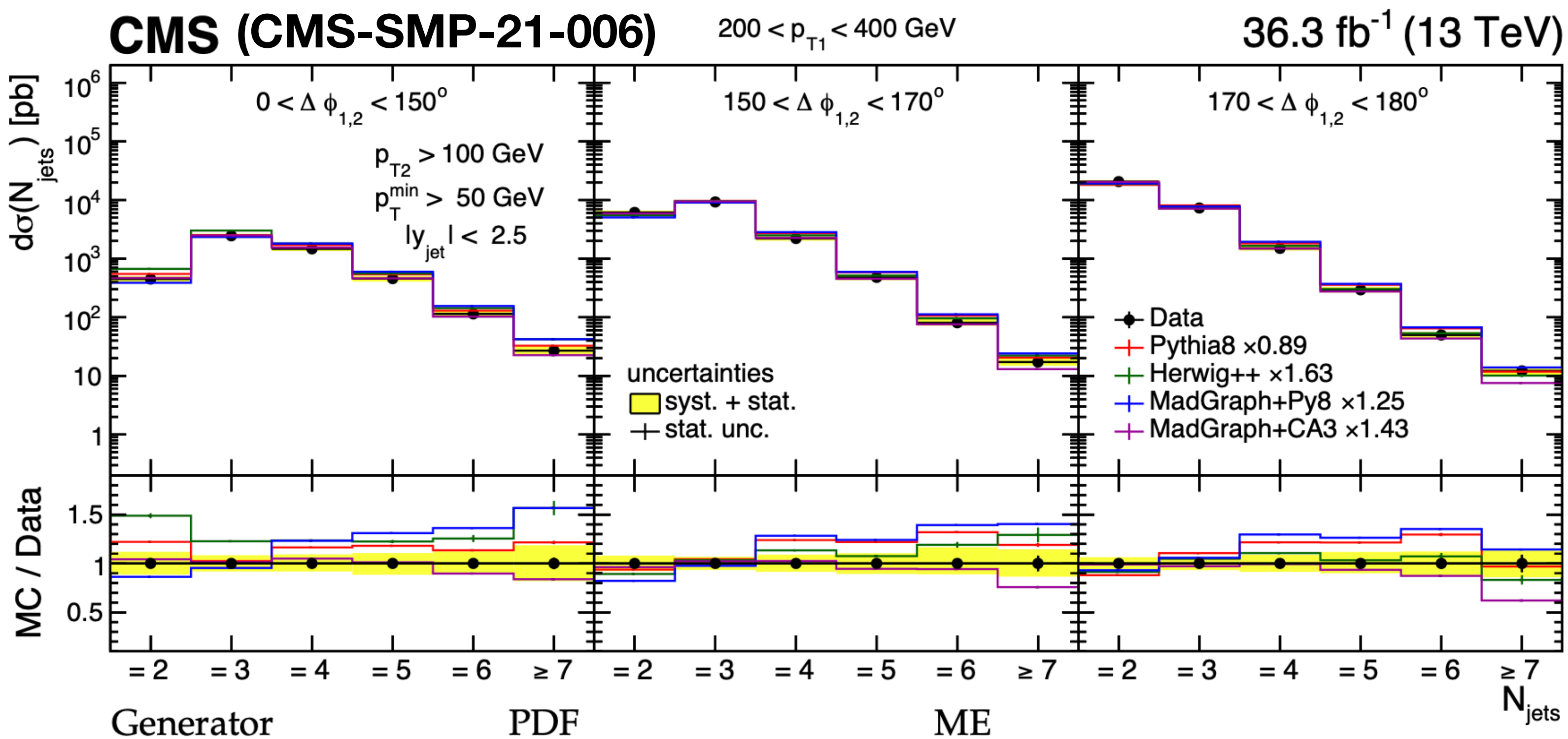
Unsupervised and lightly supervised learning in particle physics (Jai Bardhan et.al. arXiv:2403.13676)





Importance of Precision

- For Machine Learning which requires "**training**", **the big amount of well-understood data is necessary.**



Generator	PDF	ME
PYTHIA8 [23]	NNPDF 2.3 (LO) [25]	LO $2 \rightarrow 2$
MADGRAPH+PY8 [4]	NNPDF 2.3 (LO) [25]	LO $2 \rightarrow 2,3,4$
MADGRAPH+CA3 [4]	PB-TMD set 2 (NLO) [1]	LO $2 \rightarrow 2,3,4$
HERWIG++ [26]	CTEQ6L1 (LO) [27]	LO $2 \rightarrow 2$

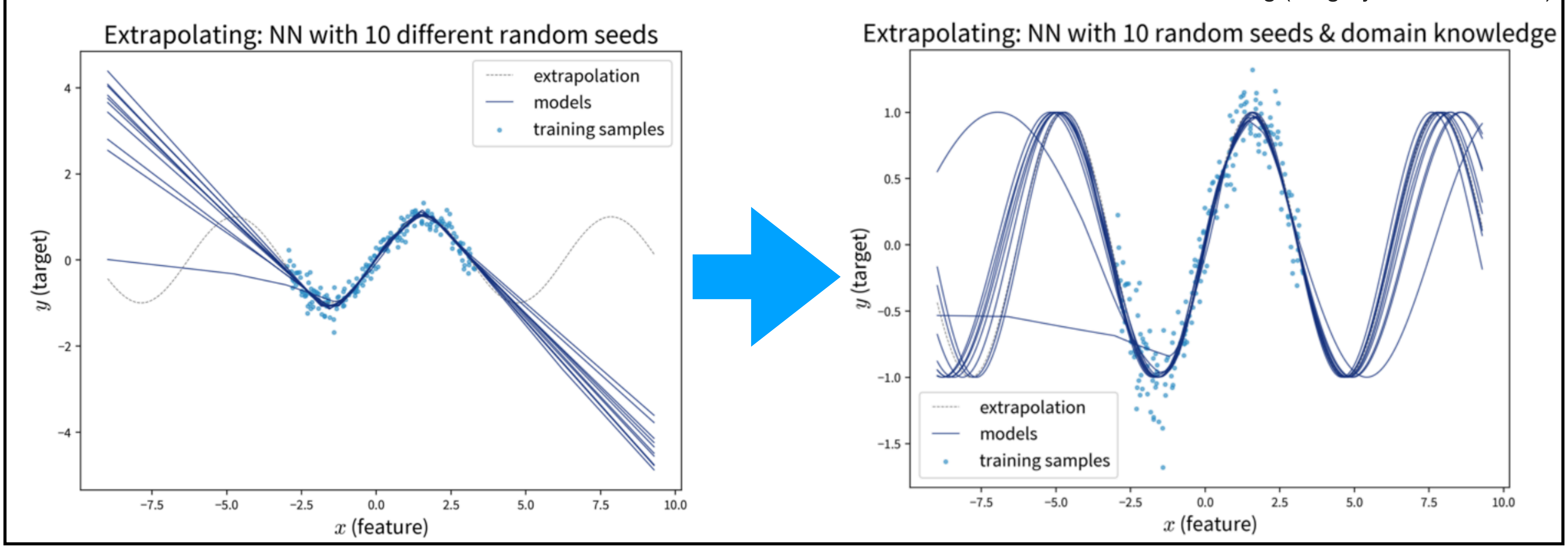
Importance of Theory

- We need **HUGE "training data"** to feed the **"data hungry" Neural Net.**

- One can dream of "data-driven" machine learning.

  \- We cannot guarantee the estimation out of Controlled samples.
  : **NO magic can do "Exploration".**
  : Domain knowledge is strongly required.

A Practitioner's Guide to Machine Learning (Blog by Franziska Horn)



Region of the "LOW" statistics

- As we get a statistics,
  we are approaching a high energy region
  = **Huge multiplicity.**

$gg \to gg$

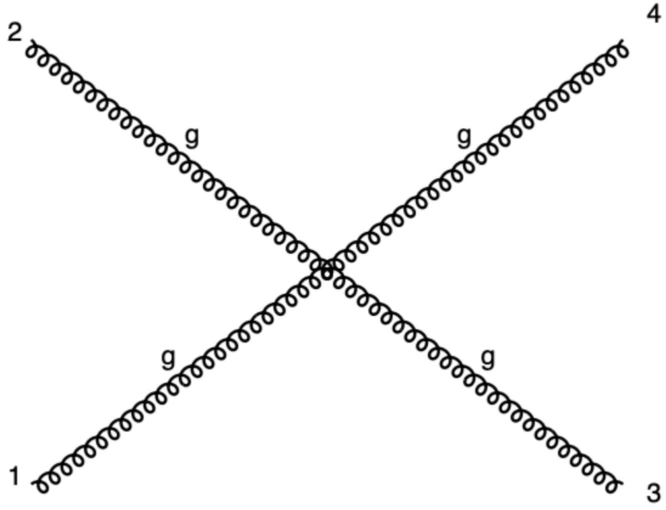


diagram 1 QCD=2, QED=0

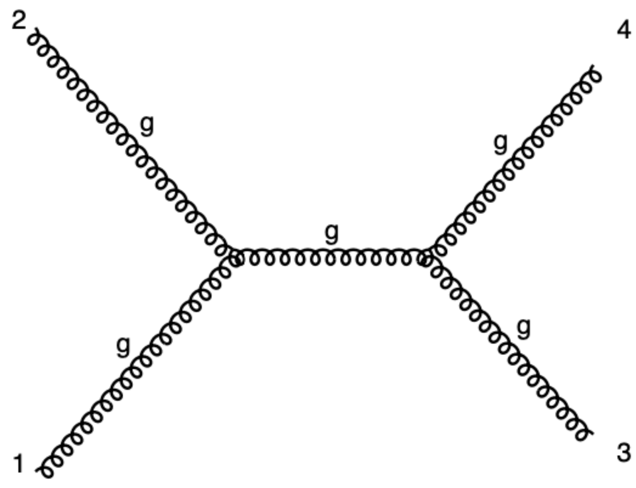


diagram 2 QCD=2, QED=0

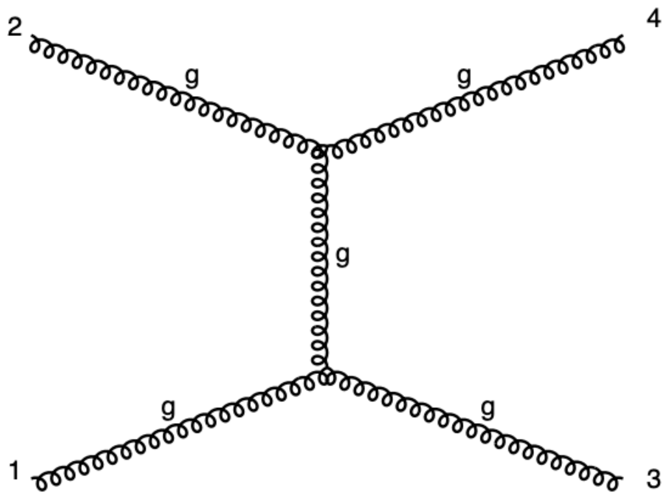


diagram 3 QCD=2, QED=0

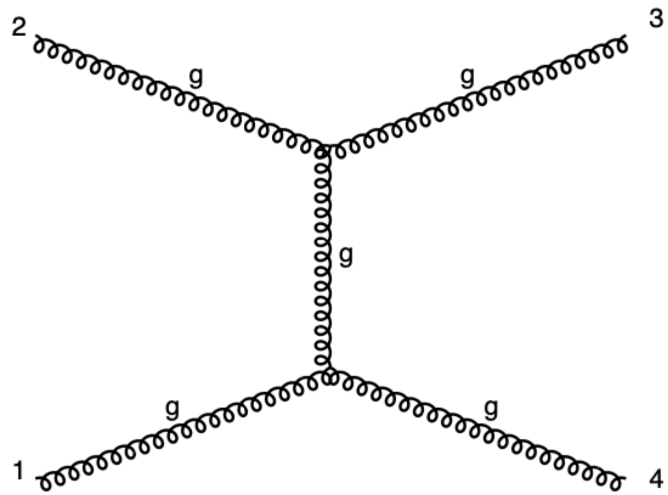
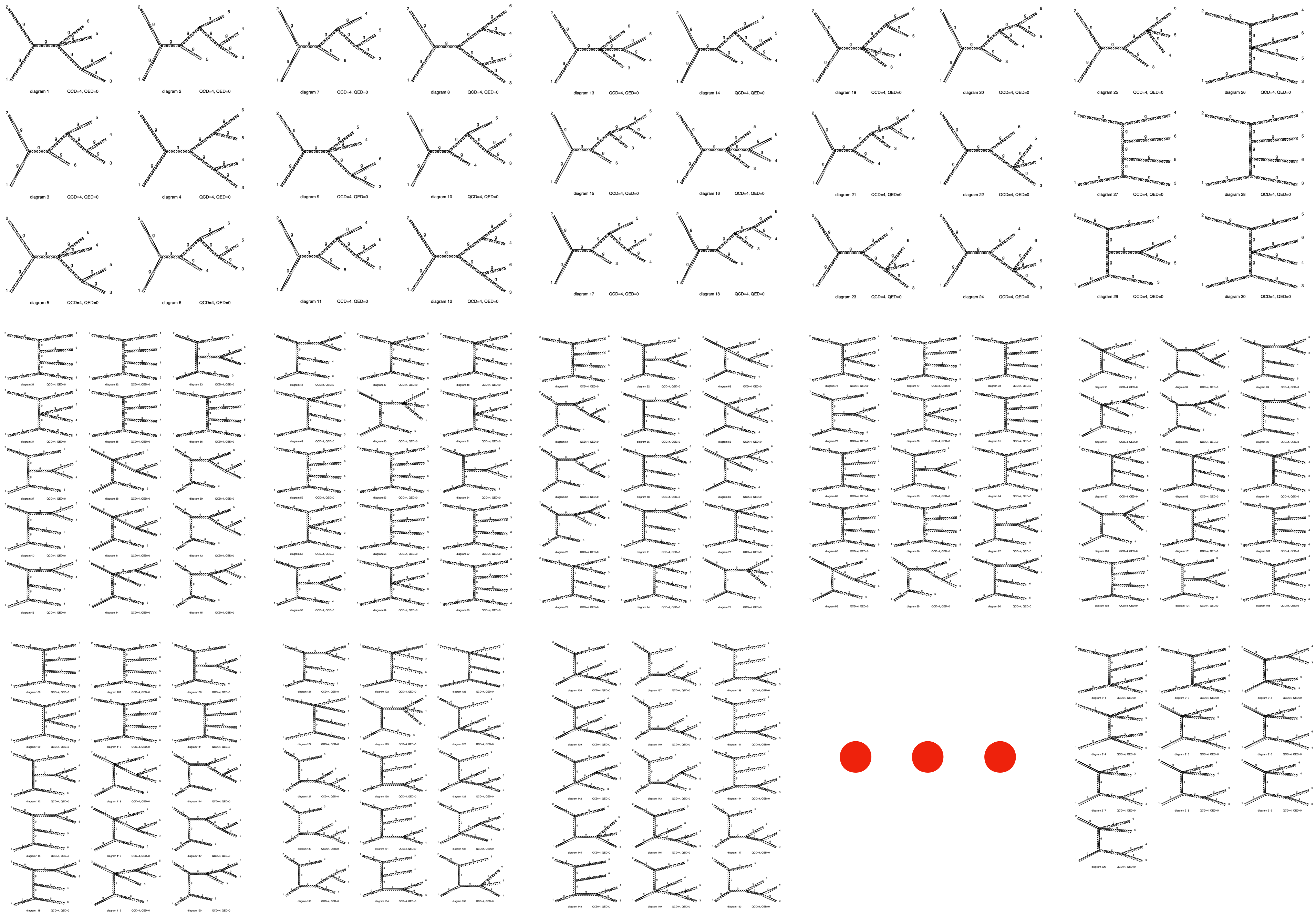


diagram 4 QCD=2, QED=0

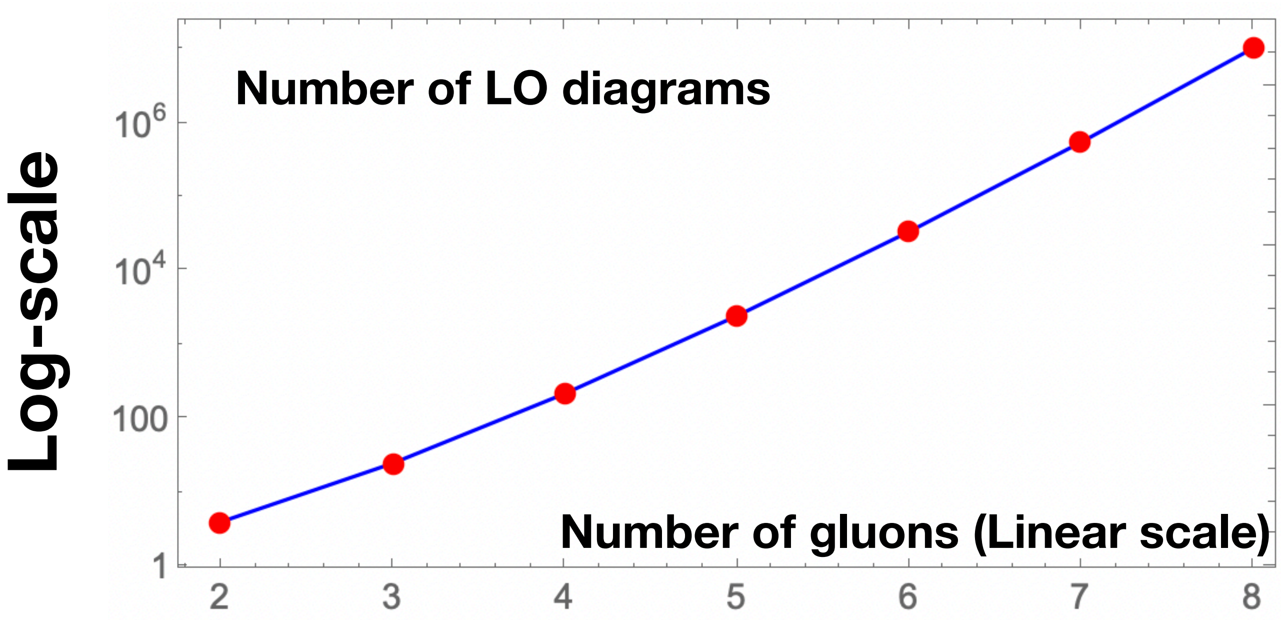
$gg \to gggg$



Multiplicity !

- The production of multi-particles will have **HUGE number** of "feynman" diagrams.

	$gg \to 2g$	$gg \to 3g$	$gg \to 4g$	$gg \to 5g$	$gg \to 6g$	$gg \to 7g$	$gg \to 8g$
Number of LO diagrams	4	25	220	2,485	34,300	559,405	10,525,900

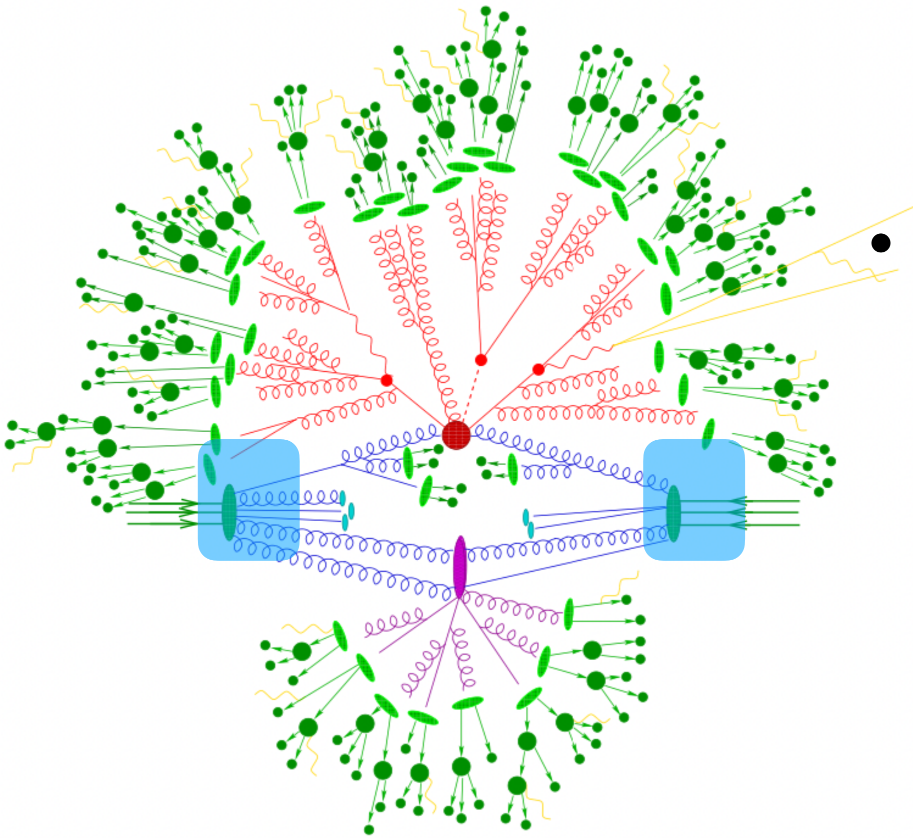


A Monte Carlo Generator

- Monte Carlo Generator is the essence of our understanding in various parts of "particle physics".
- Perturbative calculations, non-perturbative regions, Quantum Corrections.

Monte Carlo Simulation

$$\mathscr{L}_{\text{theory}} \ni -\frac{1}{4}G^{a}_{\mu\nu}G^{a,\mu\nu} - \frac{1}{4}W^{i}_{\mu\nu}W^{i,\mu\nu} + \bar{q}\left(i\gamma^{\mu}D_{\mu} - m\right)q$$

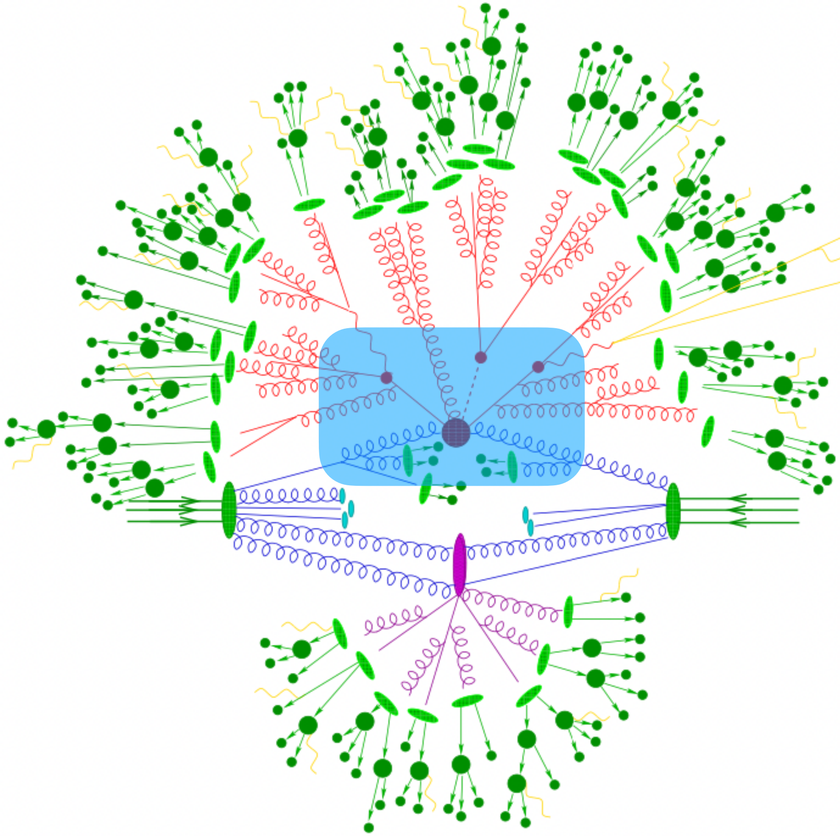


- With $\mathscr{L}_{\text{theory}}$, we simulate a collision process with various Monte Carlo tools.

  - **PDF : parton contributions** (e.g. : quark/gluons in protons)

Monte Carlo Simulation

$$\mathscr{L}_{\text{theory}} \ni -\frac{1}{4}G^a_{\mu\nu}G^{a,\mu\nu} - \frac{1}{4}W^i_{\mu\nu}W^{i,\mu\nu} + \bar{q}\left(i\gamma^\mu D_\mu - m\right)q$$

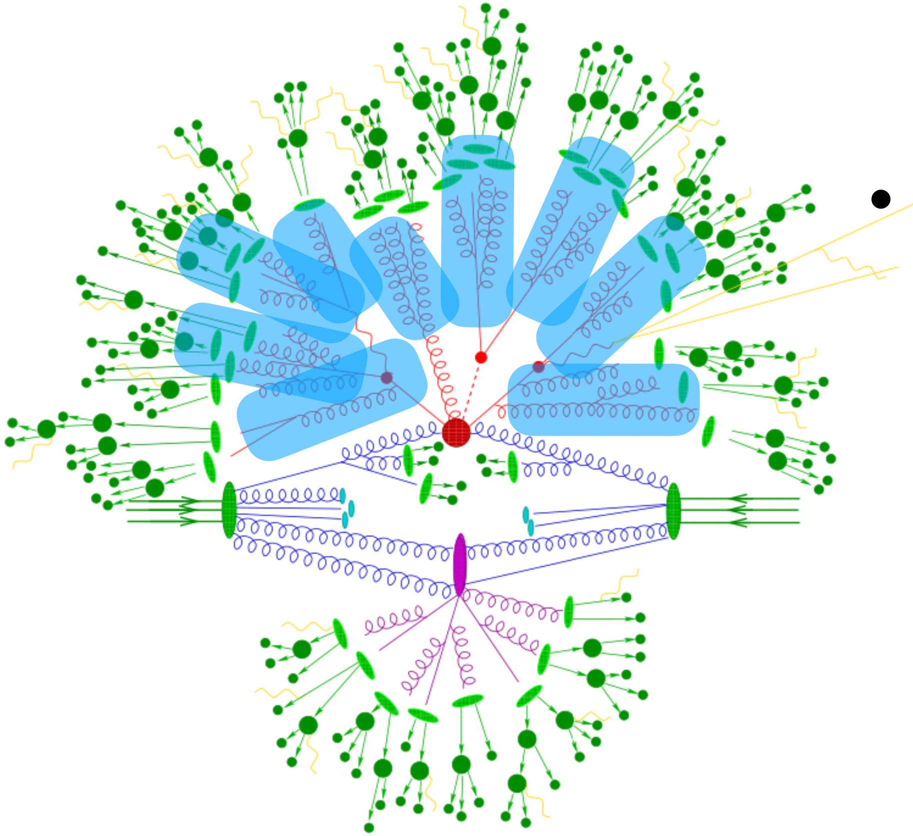


- With $\mathscr{L}_{\text{theory}}$, we simulate a collision process with various Monte Carlo tools.

  - **Hard process**
    (e.g: $gg \to t\bar{t} \to bW^+\bar{b}W^- \to b\bar{b}jjjj$)

Monte Carlo Simulation

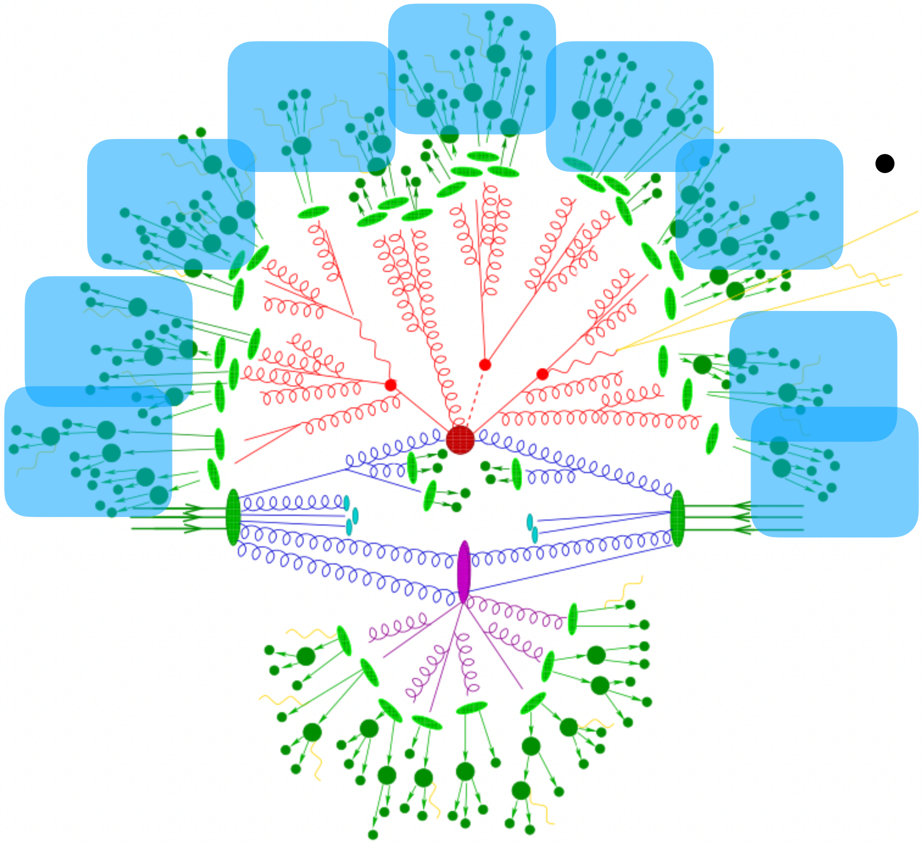
$$\mathscr{L}_{\text{theory}} \ni -\frac{1}{4}G^a_{\mu\nu}G^{a,\mu\nu} - \frac{1}{4}W^i_{\mu\nu}W^{i,\mu\nu} + \bar{q}\left(i\gamma^\mu D_\mu - m\right)q$$



- With $\mathscr{L}_{\text{theory}}$, we simulate a collision process with various Monte Carlo tools.

  - **Parton Showering** (soft radiations of charged particles)

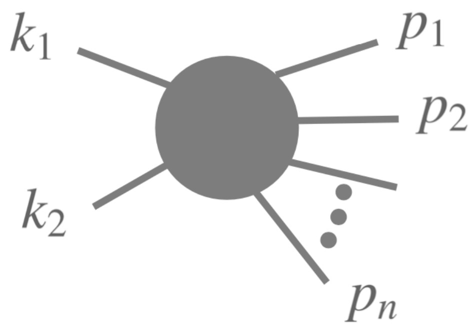
Monte Carlo Simulation



- With $\mathscr{L}_{\text{theory}}$, we simulate a collision process with various Monte Carlo tools.

- **Hadronization** **(approximation)**
(color dress-up : meson, hadron)
and **corresponding decays**

Our **First** target for



- $$\sigma = \frac{1}{2s}\int \prod_{i=1}^{n} \frac{d^3p_i}{(2\pi)^3 2E_i}\delta\left(k_1 + k_2 - \sum_{i=1}^{n} p_i\right) |M_{fi}|^2 \theta_{\rm cut}(p1, \cdots, p_n)$$

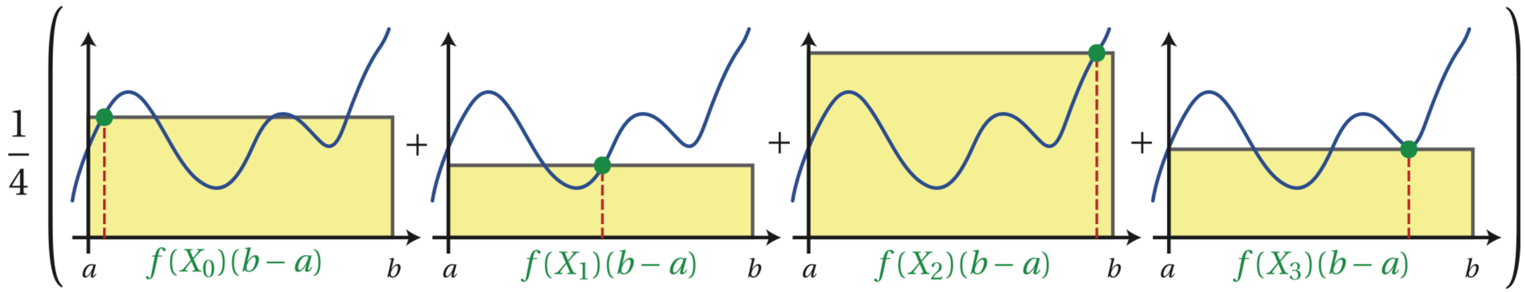
- For an observable $O(p_1, \cdots, p_n)$, we need to calculate the differential distribution of

$$\frac{d\sigma}{dO} = \frac{1}{2s}\int \prod_{i=1}^{n} \frac{d^3p_i}{(2\pi)^3 2E_i}\delta\left(k_1 + k_2 - \sum_{i=1}^{n} p_i\right) |M_{fi}|^2 \theta_{\rm cut}(p1, \cdots, p_n)\delta(O - O(p1, \cdots, p_n))$$

(precise numerical) Integration in high dimensional phase space

Monte Carlo with Importance sampling

- With random N samples according to a uniform Probability Distribution Function $\mathrm{pdf}(x)$ within an integral domain $[a, b]$



$$\mathrm{pdf}(X_i) = \frac{1}{b-a}$$

$$\langle F^N \rangle = \frac{1}{N}\sum_{i=0}^{N} \frac{f(X_i)}{\mathrm{pdf}(X_i)} = (b-a)\frac{1}{N}\sum_{i=0}^{N} f(X_i)$$

$$\rightarrow E[\langle F^N \rangle] = (b-a)\frac{1}{N}\sum E[f(X_i)] = (b-a)\frac{1}{N}\sum \int_a^b f(X_i)\mathrm{pdf}(x)dx = \int_a^b f(x)dx$$

$$\sigma[\langle F^N \rangle] = \frac{1}{\sqrt{N}}\sqrt{\sum_{i=0}^{N-1} \sigma^2\left[\frac{f(X_i)}{\mathrm{pdf}(X_i)}\right]}$$

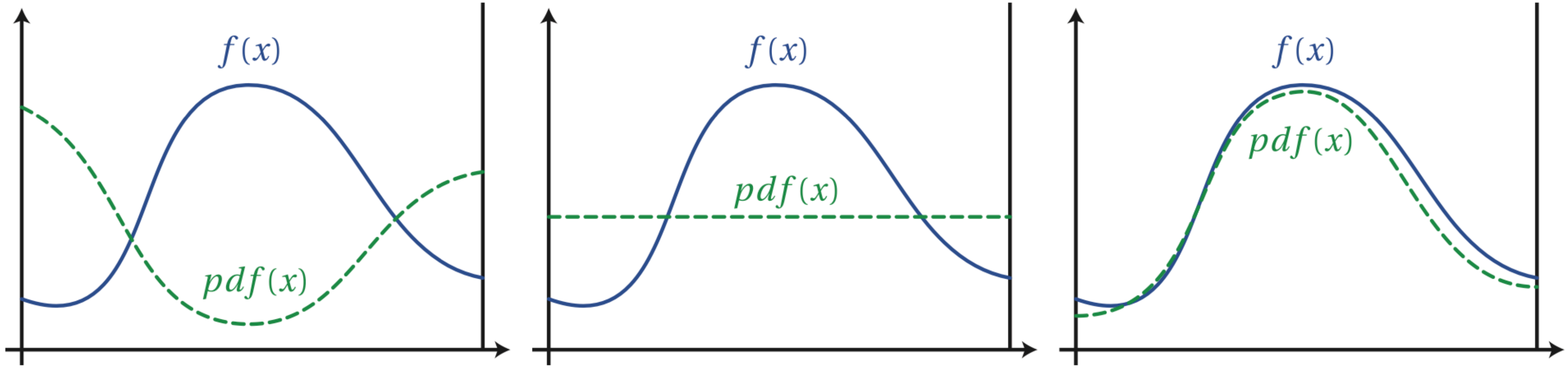
Random sampling

Importance sampling to reduce a variance

Importance sampling

- If we sample PDF $\propto f(x)$, we can reduce a variance

$$\sigma[\langle F^N \rangle] = \frac{1}{\sqrt{N}} \sqrt{\sum_{i=0}^{N-1} \sigma^2 \left[\frac{f(X_i)}{\mathrm{pdf}(X_i)} \right]} \to \frac{1}{\sqrt{N}} \sqrt{\sum_{i=0}^{N-1} \sigma^2 \left[\frac{f(X_i)}{c f(X_i)} \right]} \simeq \frac{1}{\sqrt{N}} \sqrt{\sum_{i=0}^{N-1} \sigma^2 \left[\frac{1}{c} \right]} \equiv \frac{1}{\sqrt{N}} \times 0$$

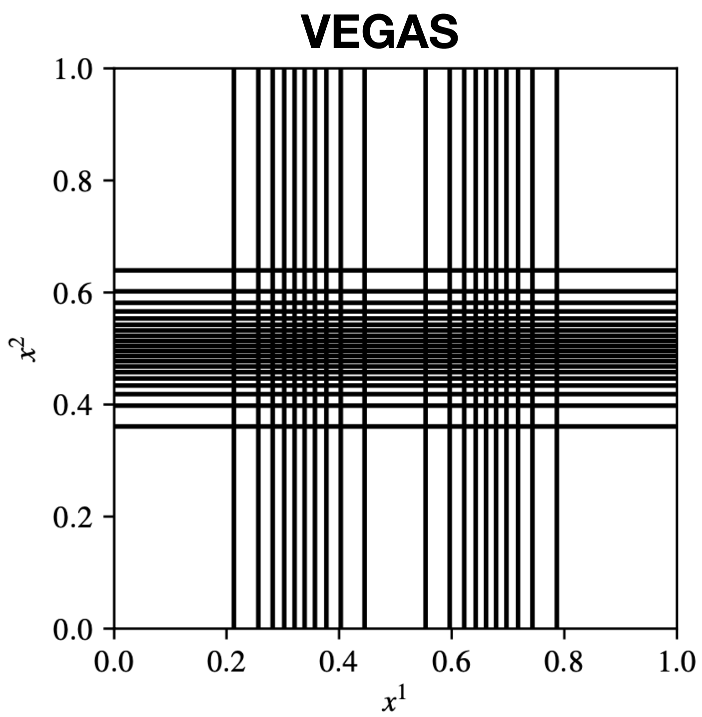


Bad PDF **Uniform PDF** **Good PDF**

- When we don't know a function $f(x)$ at all, how can we estimate a good PDF ?

Traditional method...

- **Stratified Sampling**: Divide domain **into sub-domains**.
  For example, if we divide the domain into N divisions,
  $\sigma \propto \frac{1}{N}$ instead of $\sigma \propto \frac{1}{\sqrt{N}}$



- "Classic" VEGAS: **Adaptive** importance sampling, since 1977

  Recently, there is an update, VEGAS+
  J.Comput.Phys. 439 (2021) 110386

Neural Net (NN) as a good estimator

- Due to the universal approximation theorem, **NN serves as a bonafide function approximator.**

- Design a process where the accuracy of NN becomes proportional to our interests in sampled regions:

  - spend, relatively, more time to sample in **regions of interests**

  - enough time for low importance region

Importance sampling with Machine Learning

- In fact, we already solved a similar problem in our previous study arXiv:2207.09959, (This project was a pilot project)

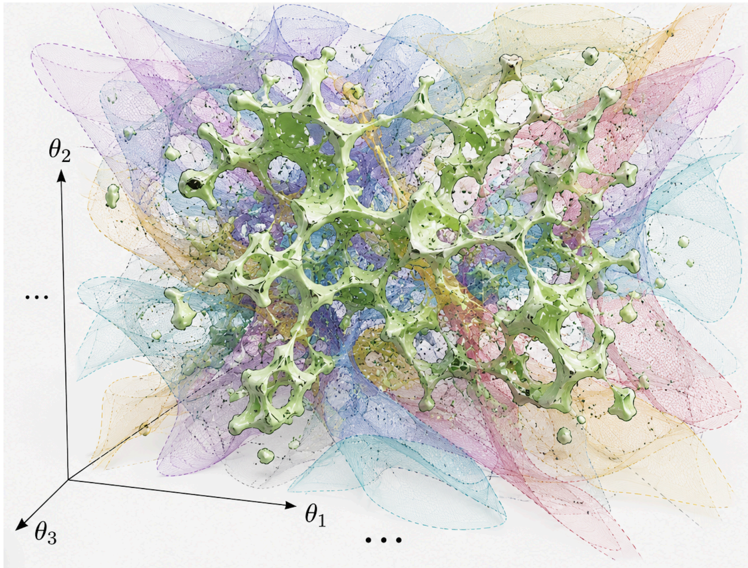
  "Exploration of Parameter Spaces Assisted by Machine Learning" (Computational Physics Communication, v293, 2023)

- Let me explain what we have done....

Conventional phenomenology

$\mathscr{L}$

Theory model
(a framework)



Precision measurements from collider, flavor, dark matter and cosmology

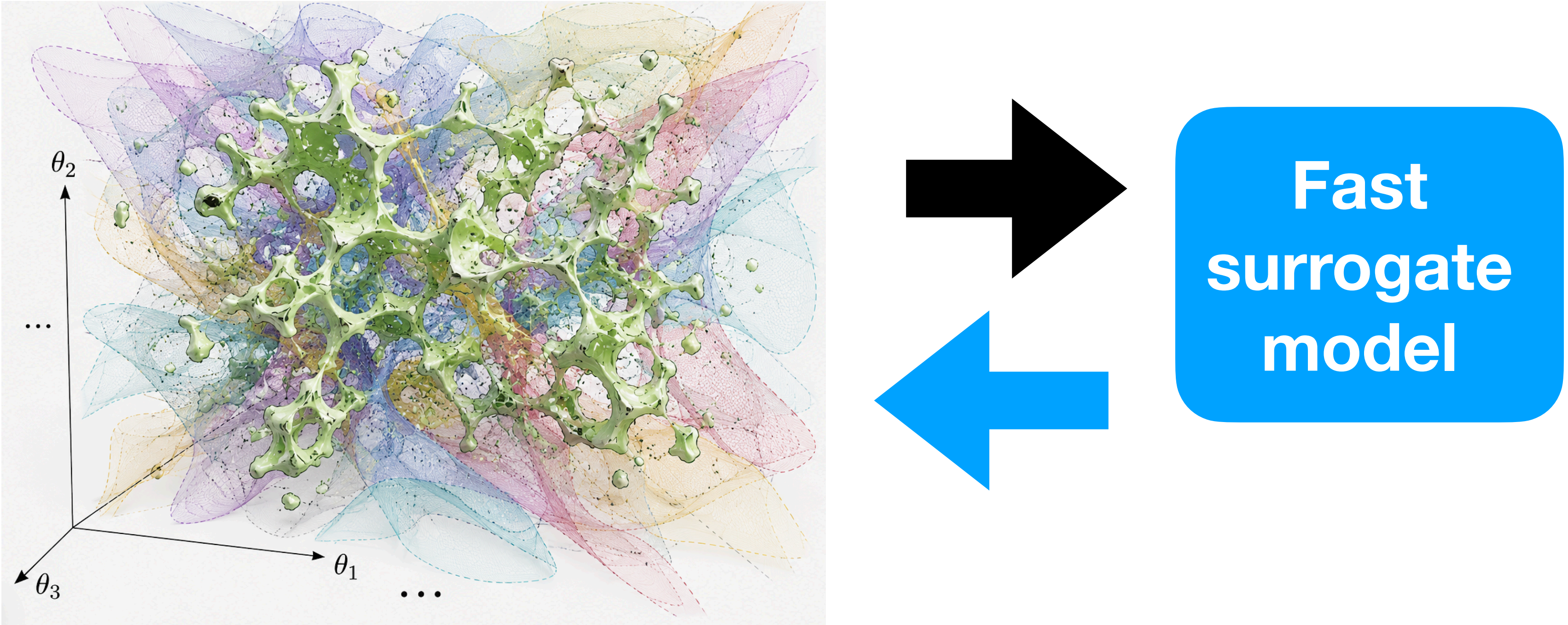
Time consuming numerical calculations

Numerical calculators (Monte-Carlo, relic...)

Highly constrained and irregular surviving parameter space

- Theoretical models often contain many free parameters.
- Precision measurements exclude large portions of the parameter space.
- The surviving parameter space is highly irregular, making brute-force scans increasingly inefficient.

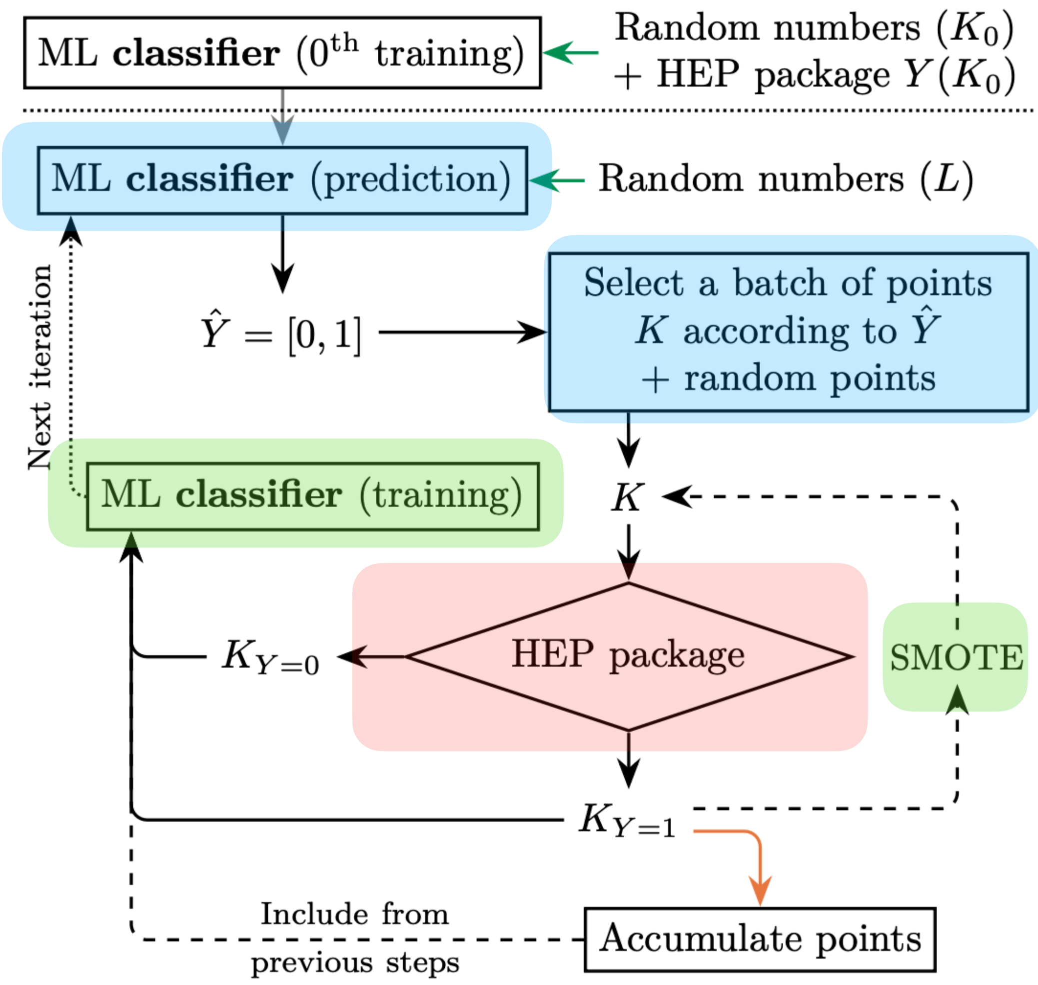
AI assisted phenomenology



- We utilize AI to **approximate a heavy calculator** precisely
  - to reduce a time

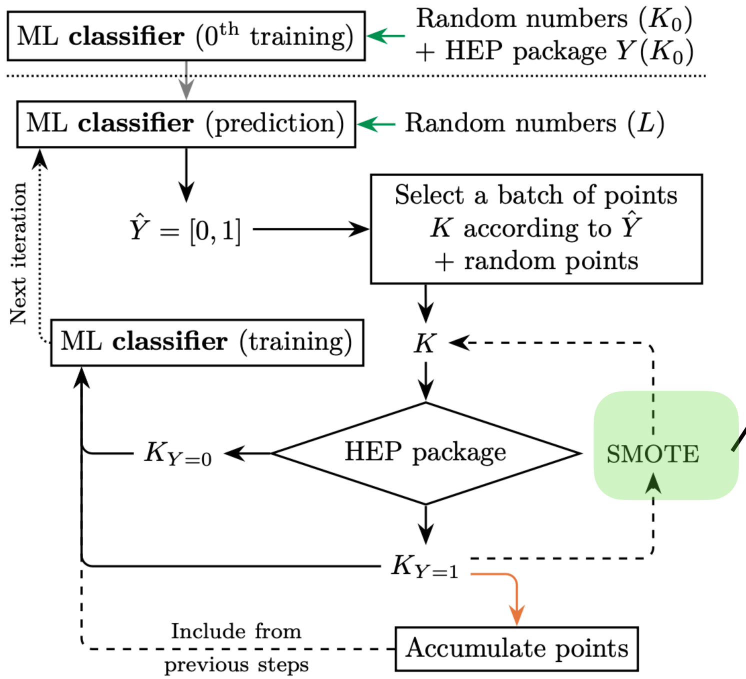
- and to **suggest regions of interest**, not to waste a valuable time over unnecessary / uninteresting parameter regions

Classifier type ML



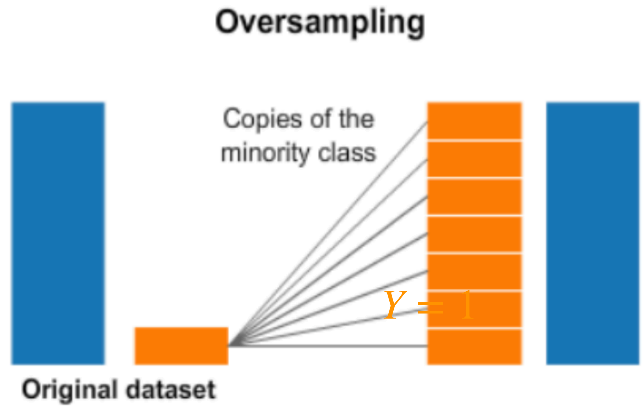
- **Generate initial random points**
  - Evaluate them with a HEP simulation package.
- **Train a classifier**
  - Learn whether a point is accepted $\hat{Y} = 1$ or rejected $\hat{Y} = 0$
- **Predict promising regions**
  -Apply the classifier to a large pool of unexplored points.
- **Select and evaluate new candidates**
  - Focus on points likely to satisfy the target conditions.
- **Update and repeat**
  - Add newly evaluated points to the dataset and retrain the classifier.

Data Augmentation using SMOTE



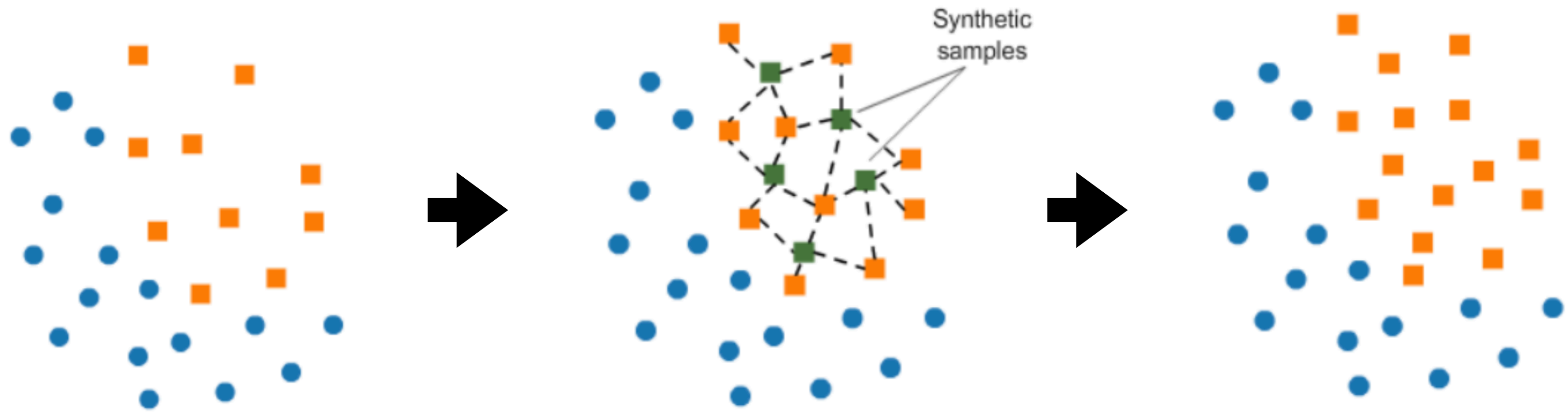
SMOTE (arXiv:1106.1813)

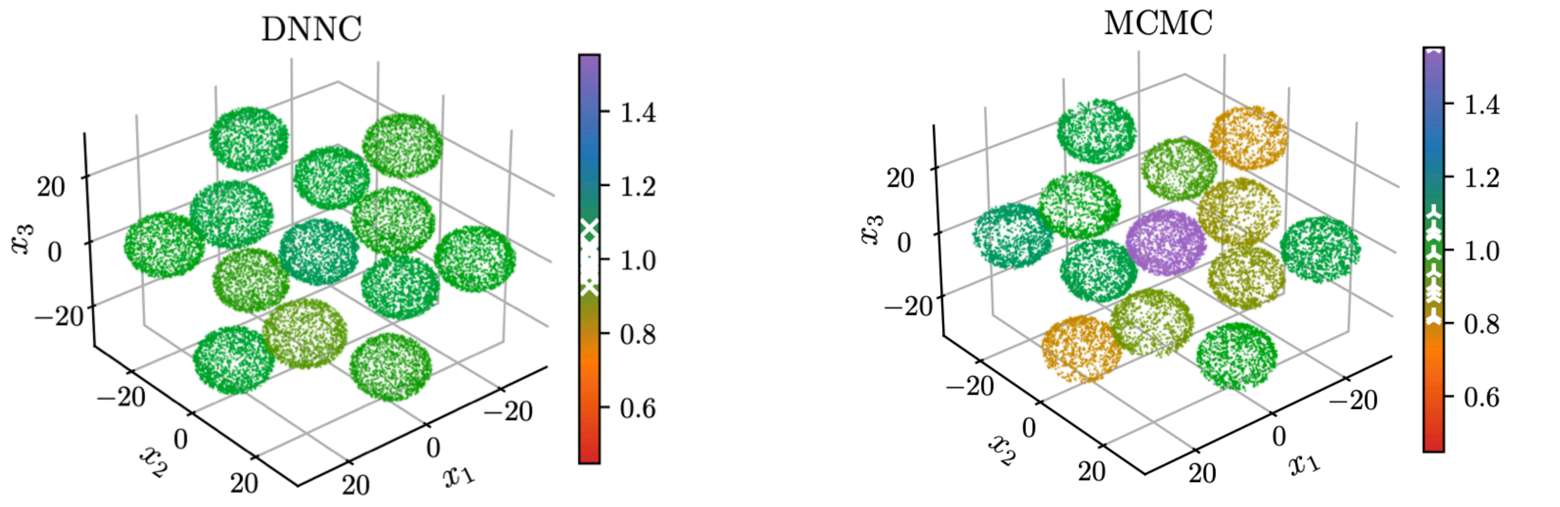
- **S**ynthetic **M**inority **O**versampling **T**echnique



SMOTE generates artificial minority-class events by interpolating between nearby signal samples in feature space.

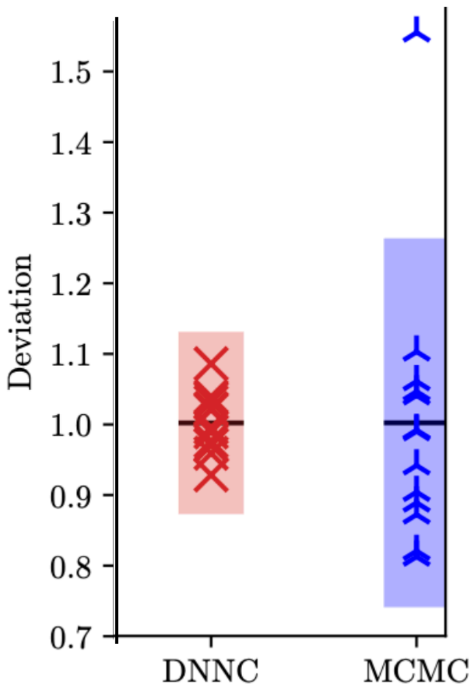
Generate synthetic minority-class samples using k-nearest neighbors





$$O_{3\mathrm{d}} = \left[2 + \cos\left(\frac{x_1}{7}\right)\cos\left(\frac{x_2}{7}\right)\cos\left(\frac{x_3}{7}\right)\right]^5$$

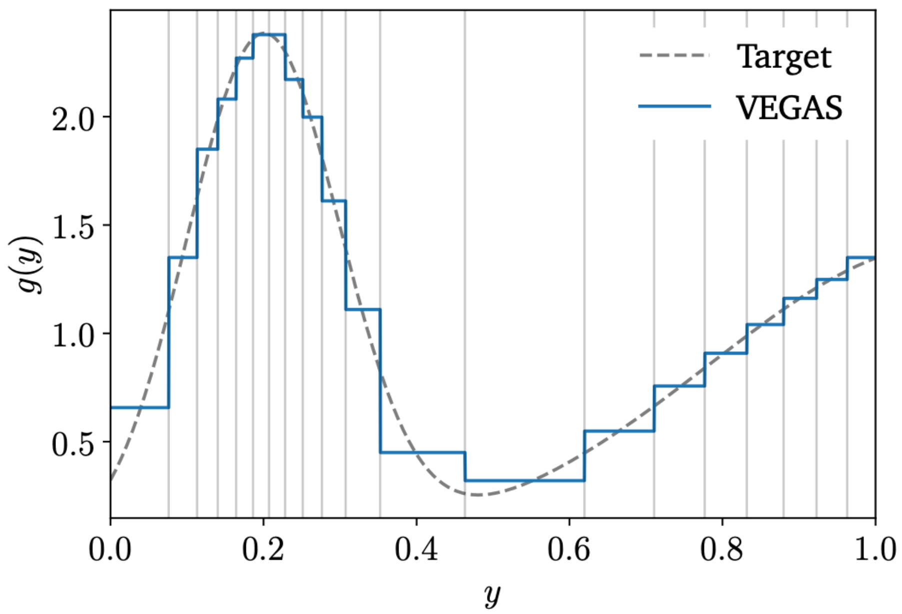
$$\mathcal{L}_{3\mathrm{d}} = \exp[-(O_{3\mathrm{d}} - c_{3\mathrm{d}})^2/2\sigma_{3\mathrm{d}}^2]$$



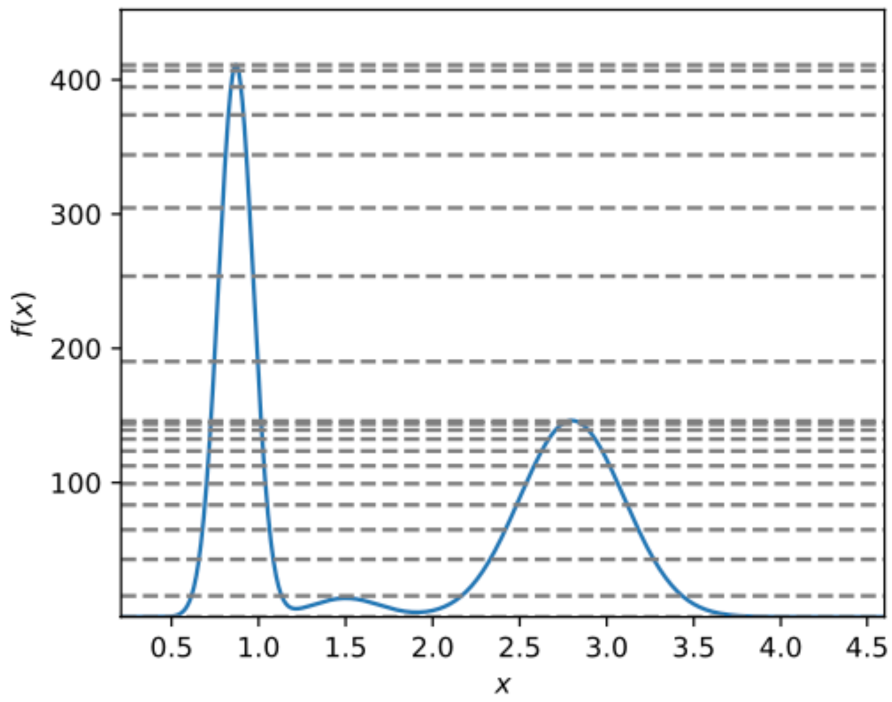
- Points are $\mathscr{L} > 0.9$ conditions.
- With $x_i \in [-10\pi, 10\pi]$, there are 13 cell.
- The "**deviation**" is the ratio of **a population in each "cell"** over an average population

Utilizing our ML algorithm for an importance sampling in an integration

Two integration methods



- **Riemann** Integration.

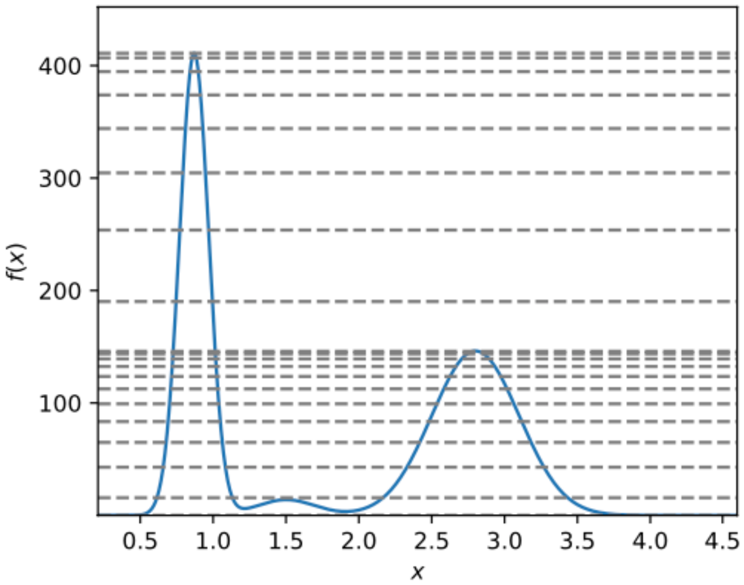


- **Lebesgue** Integration

- **Lebesgue integration provides a natural framework for importance sampling.**

  - A classical example: $f(x) = \begin{cases} 1 & \text{if } x \in P \\ 0 & \text{if } x \in Q \end{cases}$

Our approach: Lebesgue



- **Divide the space of integrand (classes)**

$$\Phi_j = \{\vec{x} \,|\, l_j < f(\vec{x}) \leq l_{j+1}\}$$

- The integral : $I_\Phi\left[f(\vec{x})\right] = \int_\Phi \mathrm{d}^d x\, f(\vec{x}) = \sum_{j=1}^{n} \int_{\Phi_j} \mathrm{d}^d x\, f(\vec{x}) = \sum_{j=1}^{n} V_{\Phi_j} \langle f \rangle_{\Phi_j}$

V_{Φ_j} : Volume of Φ_j .

- We recast the **problem of integration** → **classification problem**

example

- $\infty - \infty = \text{finite}$: We are testing "fine-tuning" function of

$$f_1(x_1, x_2) = g(x_1; 5, 2)g(x_2; 0, 2)$$

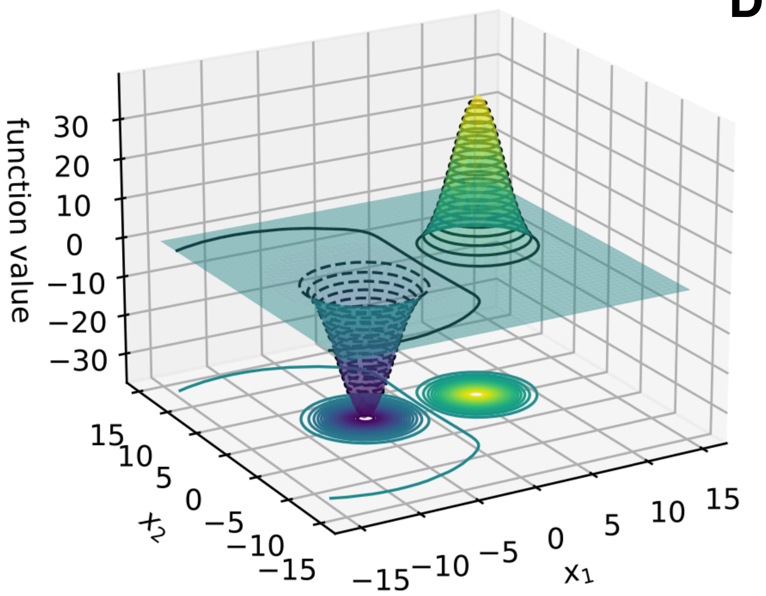
$$f_2(x_1, x_2) = g(x_1; -5, 2.1)g(x_2; 0, 2.1)$$

$$f_3(x_1, x_2) = g(x_1; 0, 3)g(x_2; 0, 3)$$

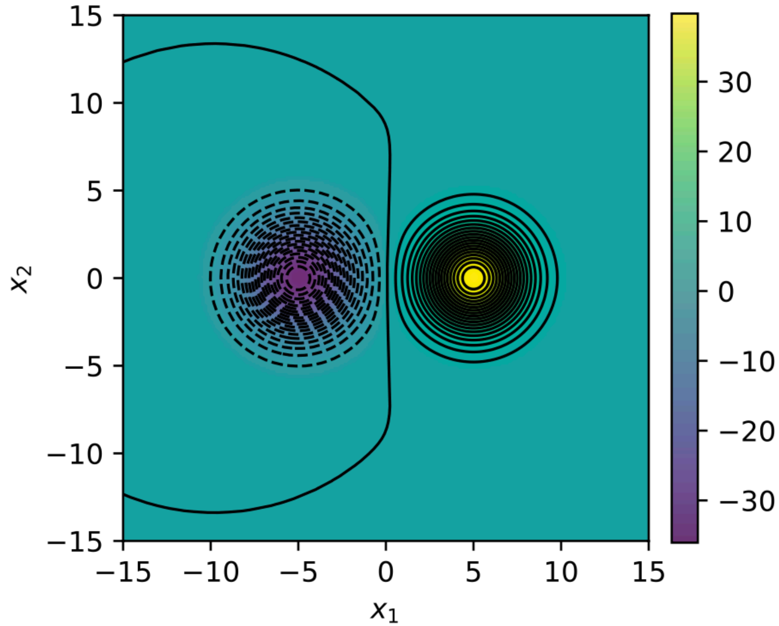
$$f(x_1, x_2) = 1000\,[f_1(x_1, x_2) - f_2(x_1, x_2)] + f_3(x_1, x_2)$$

with $g(x; m, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[\frac{(x-m)^2}{2\sigma^2}\right]$

$$\int f(\vec{x})dxdy \simeq 1$$

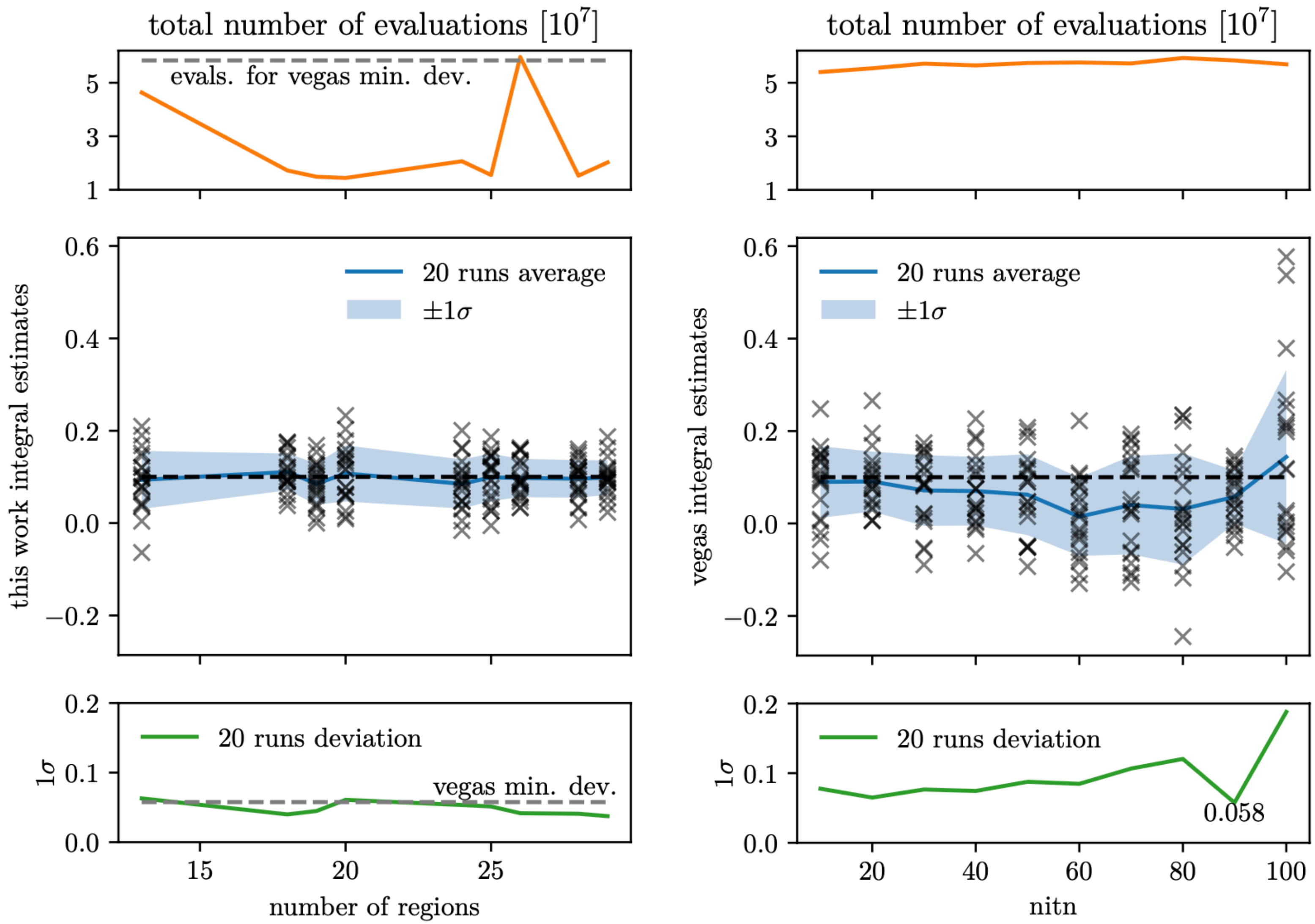


Divisions for equal "absolute" contributions



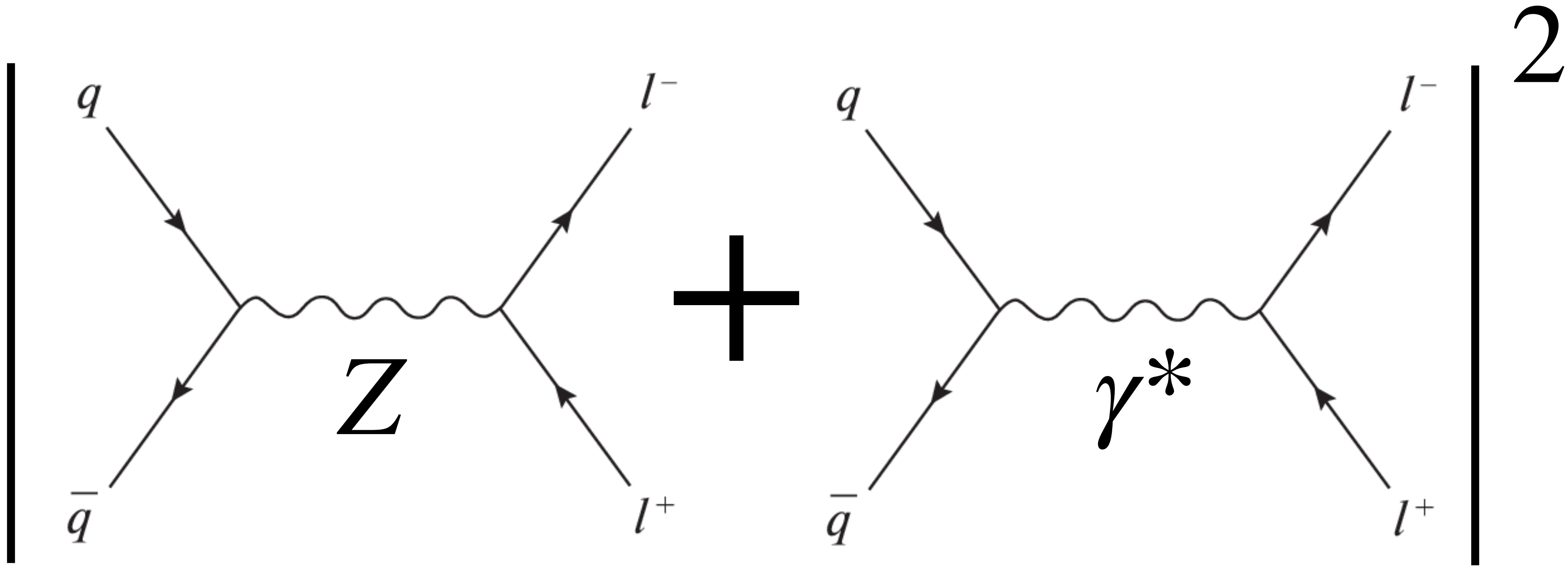
$$G(\vec{x};\vec{\mu},\sigma) = \prod_{j=1}^{7} g(x_j;\mu_j,\sigma)$$

$$f_{7D}(\vec{x}) = 100 \times [G(\vec{x};\vec{\mu}_+,\sigma_+) - G(\vec{x};\vec{\mu}_-,\sigma_-)] + 0.1 \times G(\vec{x};\vec{0},\sigma_0)$$

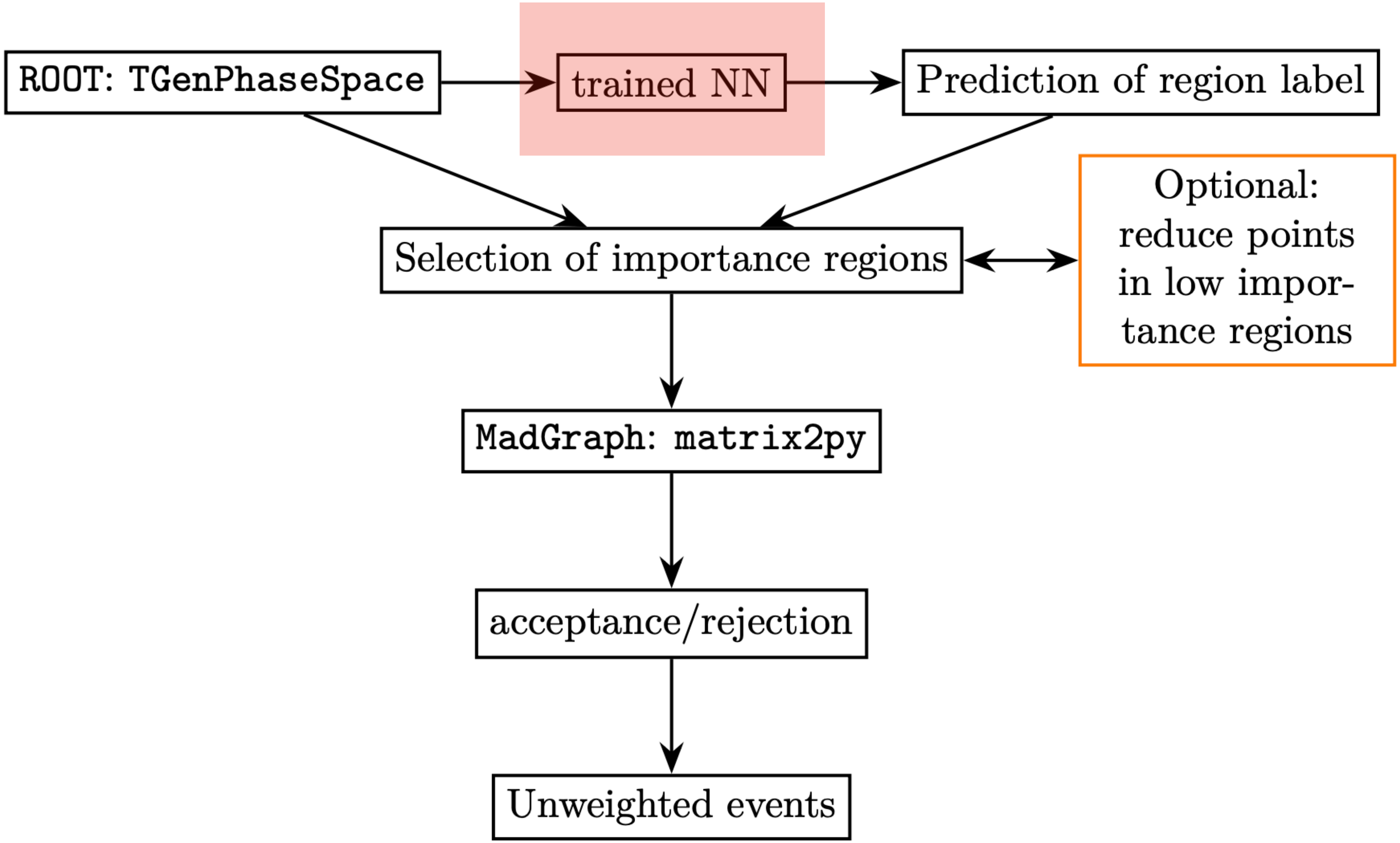


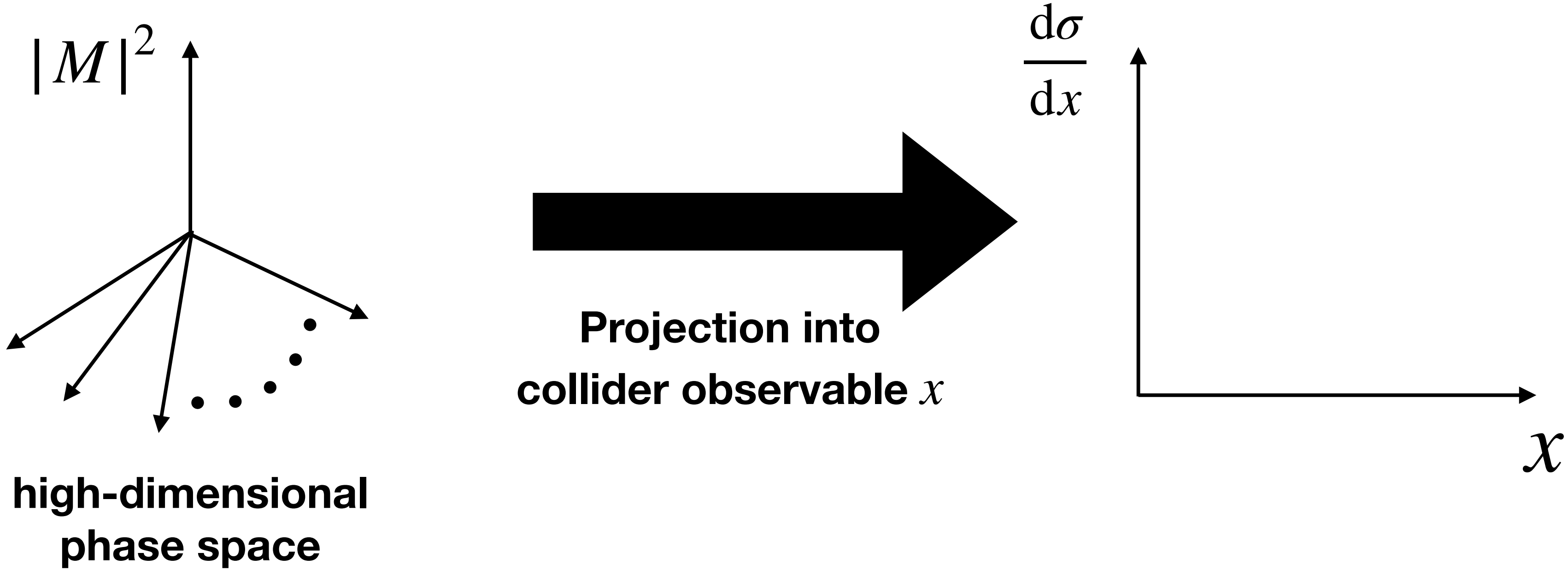
The Physics

$2 \to 2$ process with interference

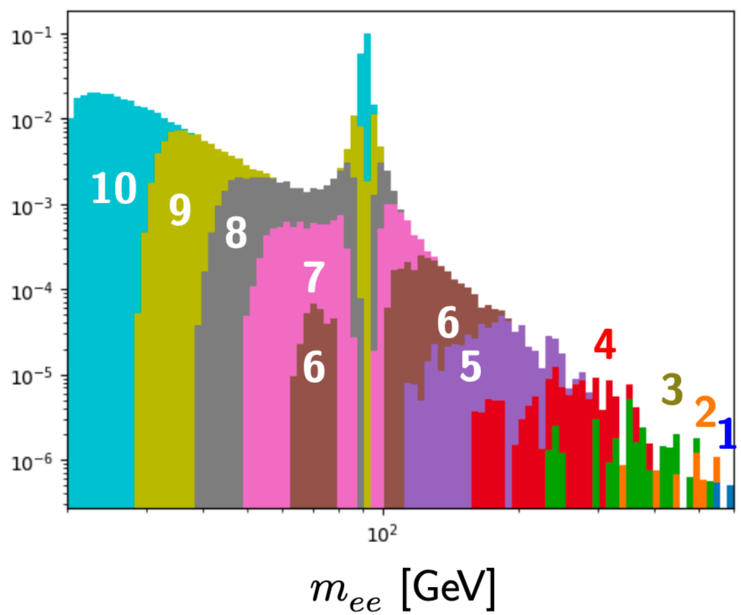


Generating MC samples with NN



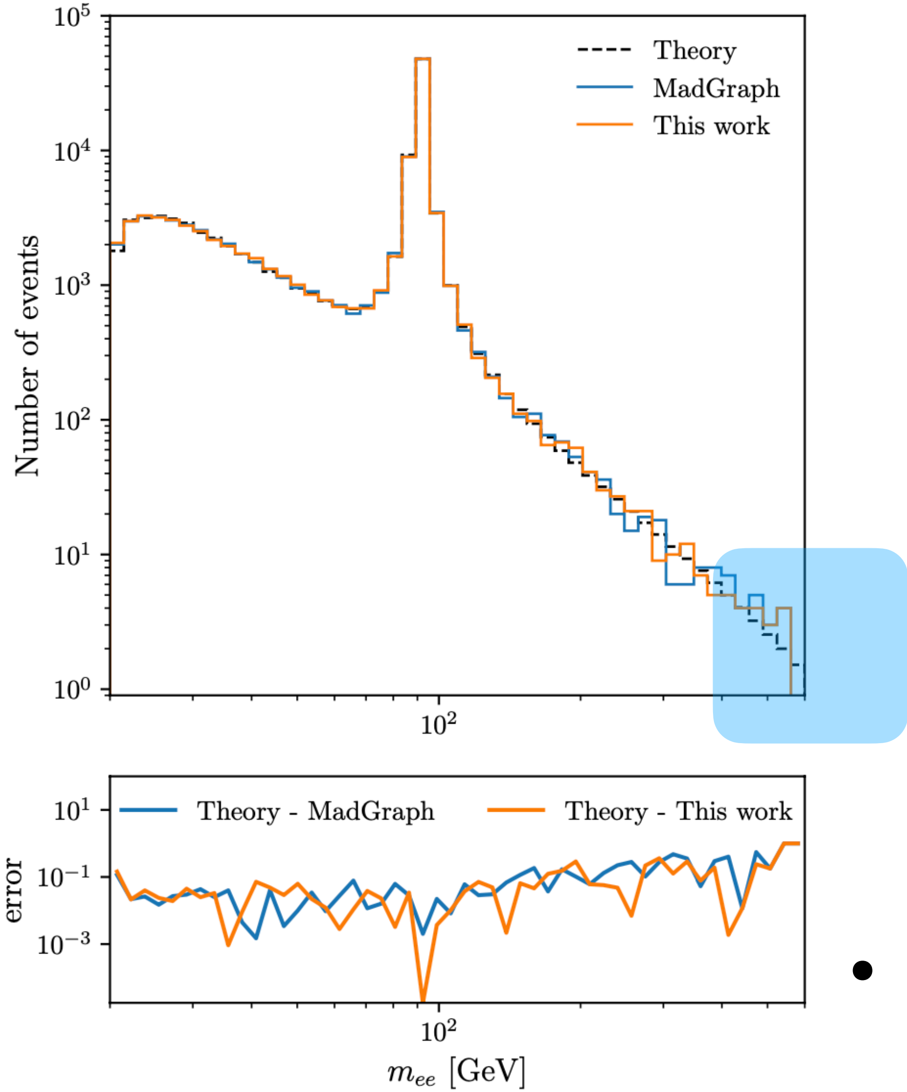


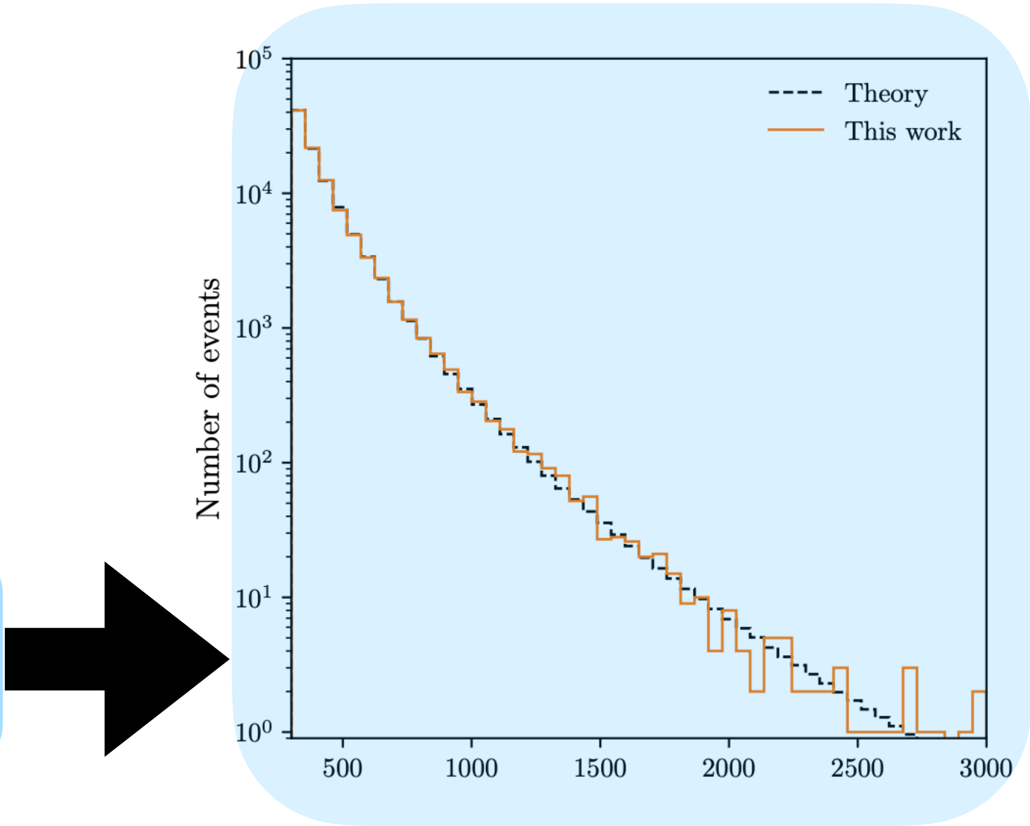
e^-e^+ invariant mass projection



- Sample each region until enough events are accumulated. **NN can tell which regions points belong to**.
- Select points using correct result.

Sample as long as we want





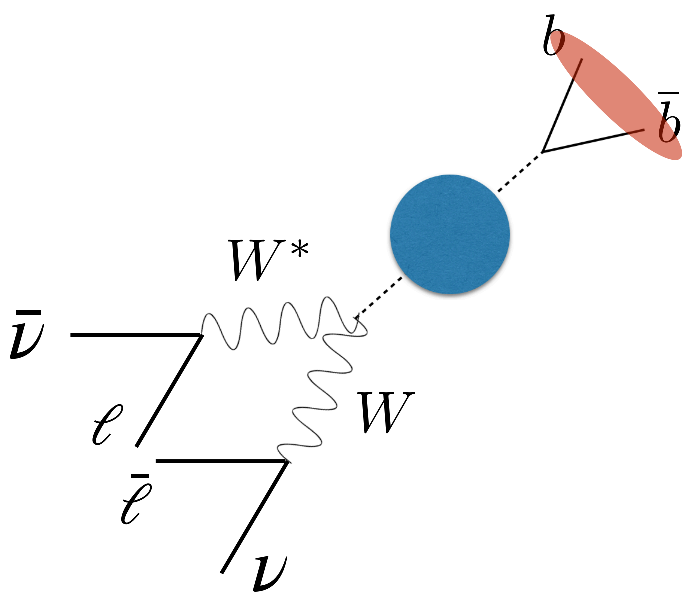
- We can **zoom into** **"rare events"**

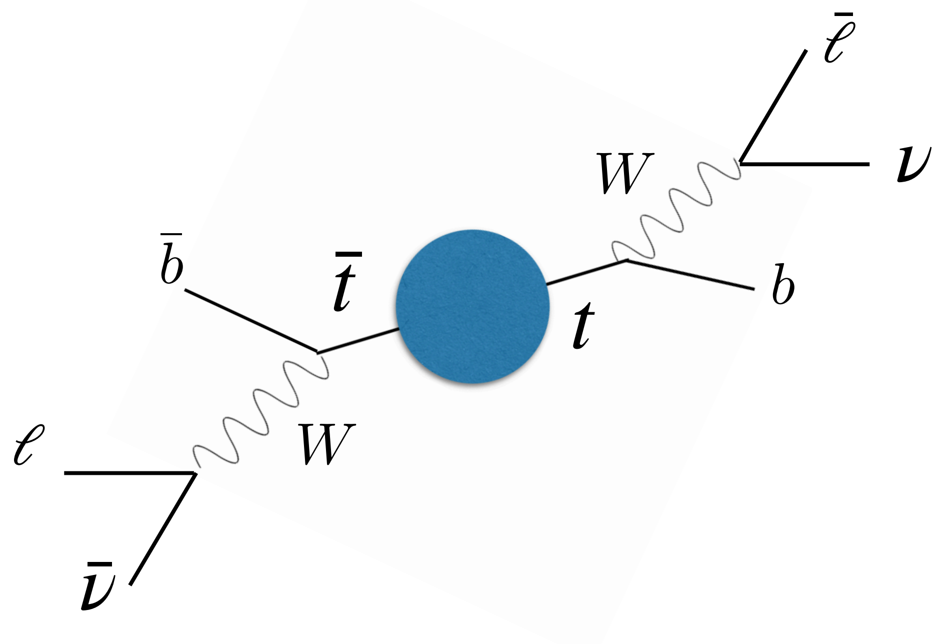
Based on the assumption: With good enough Monte Carlo

- **I will explain how one can do the "supervised" ML for the LHC analyses.**
- **Of course, we need to have an efficient (Not so much data-hungy) ML**

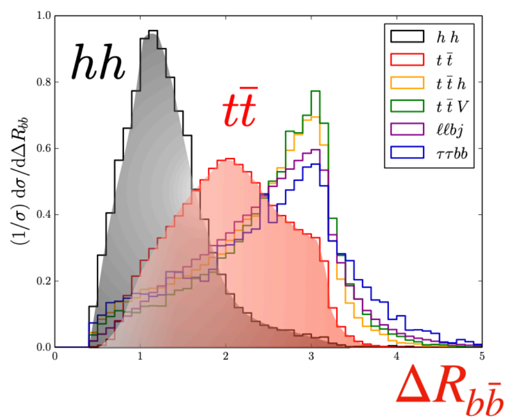
- **Parton level information: Kinematics**
- **Parton shower and hadronization: Gauge "charge"**
- **How can we maximize the combination of two different types of information?**

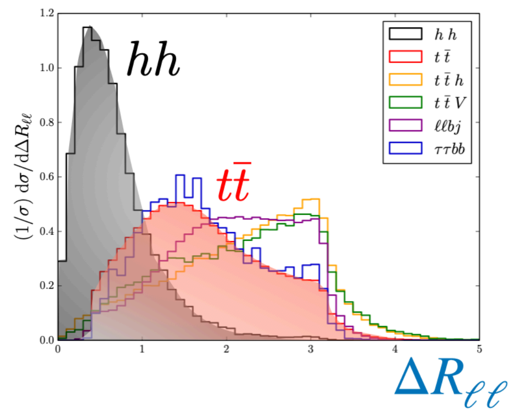
Parton level information





- **Kinematic variables** to utilize a **different phase-space** structures (signal, v.s. backgrounds)





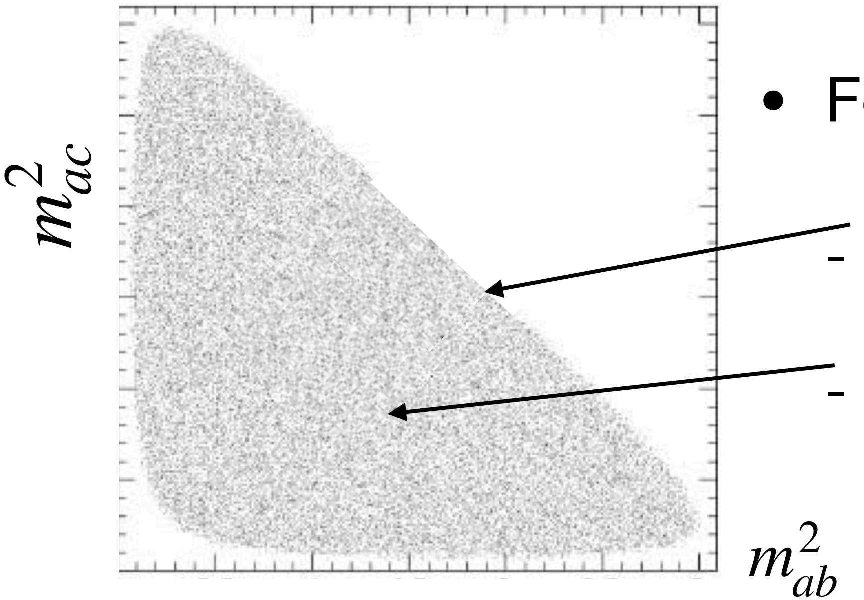
Kinematics: Global information

- Differences in kinematics are from **"high P_T" region, i.e. reconstructed level**

  - Telling us about the structure of "Feynman-diagram" (Event-topology, Mass spectrum)

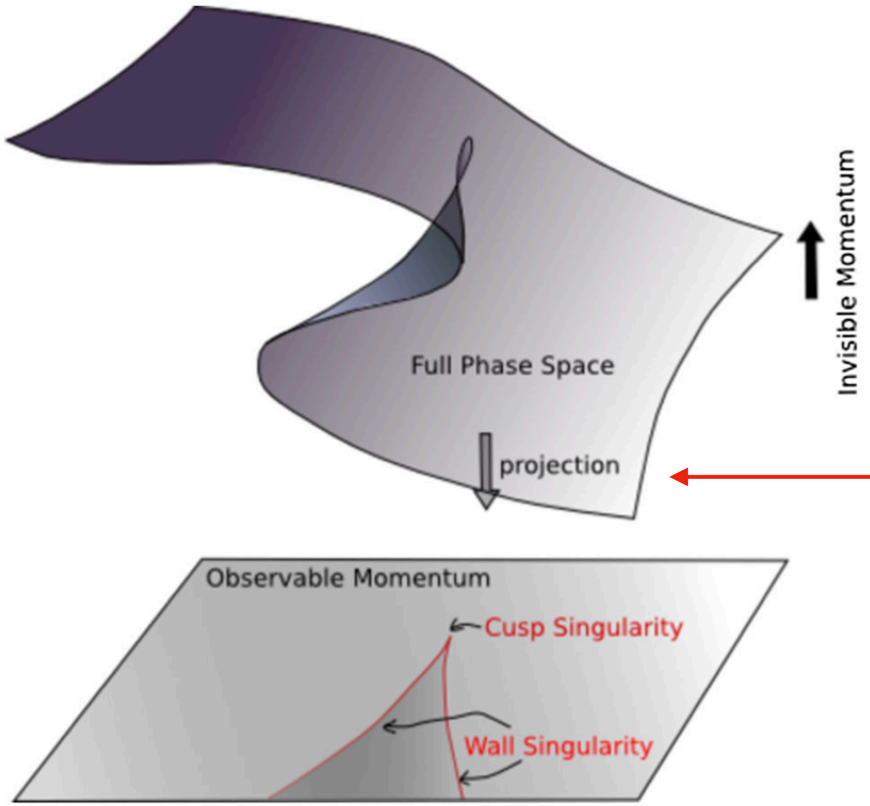
- We can further utilize $|\mathscr{M}|^2$ differences (**Density** bounded by phase-space)

Classic example: Dalitz plot



- For example, $X \to a, b, c$
  - **Boundary**: Structure of $1 \to 3$ **phase space**
  - **Density**: information on **spin**

- Constructing an "observable" from a multi-dimensional phase space is **non-trivial.** **(Especially when there is missing information)**

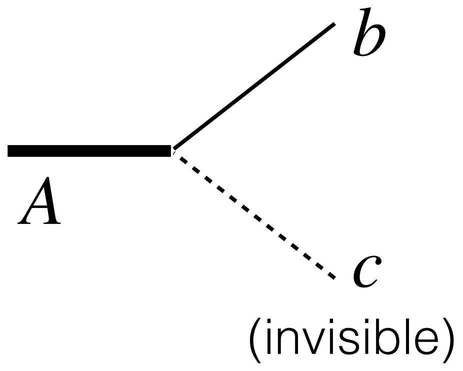


Ian-Woo Kim 2010

- Finding a suitable projection to maximize the **feature** of a full phase-space

Feature Engineering

- **Kinematic variables** to utilize a **different phase-space** structures (signal, v.s. backgrounds)



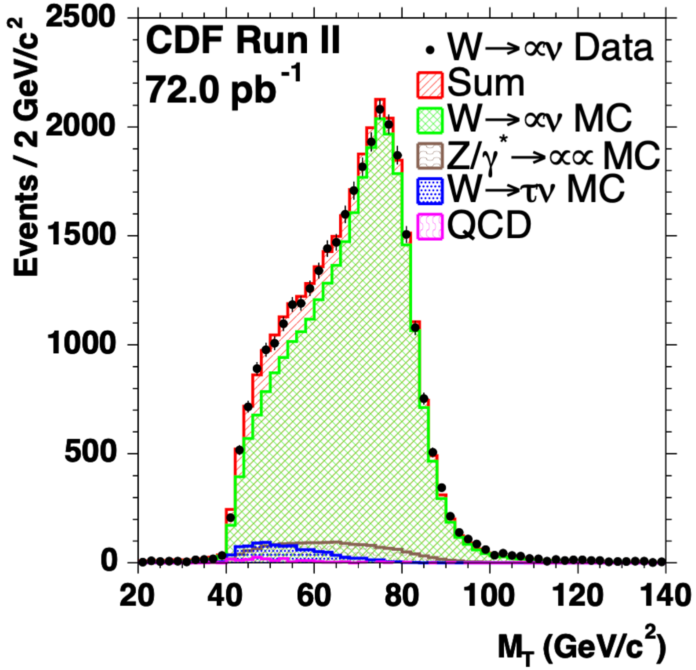
$$\theta = \{m_A\} \longrightarrow X = \{p_b^\mu, p_c^T\}$$

3: 3-Momentum from visible particle
2: Transverse Momentum from imbalanced situation

$$\dim(X) = 3 + 2 \rightarrow \dim(V) = 1$$

- **A human-engineered feature variable**, M_T which estimates m_A with an endpoint of its distribution

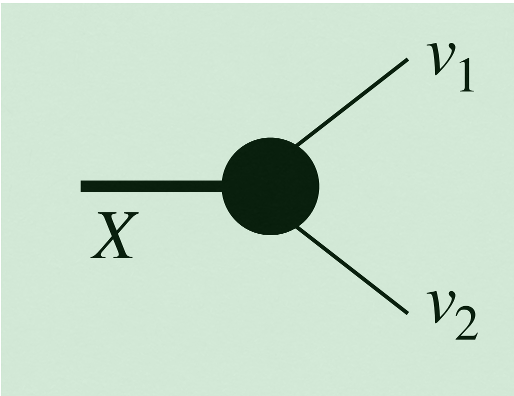
  (highly singular behavior due to its Jacobian peak)



Can AI automatically construct an optimal observable?

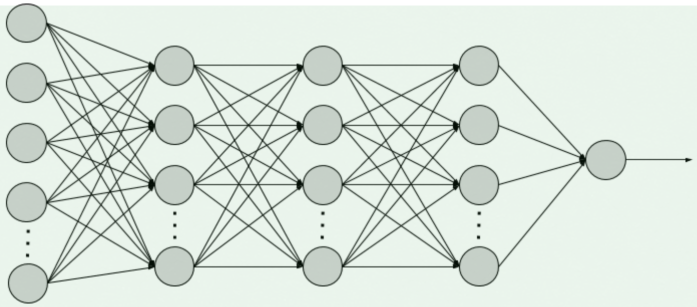
Example: Bump hunting

- **Conventional** supervised ML,



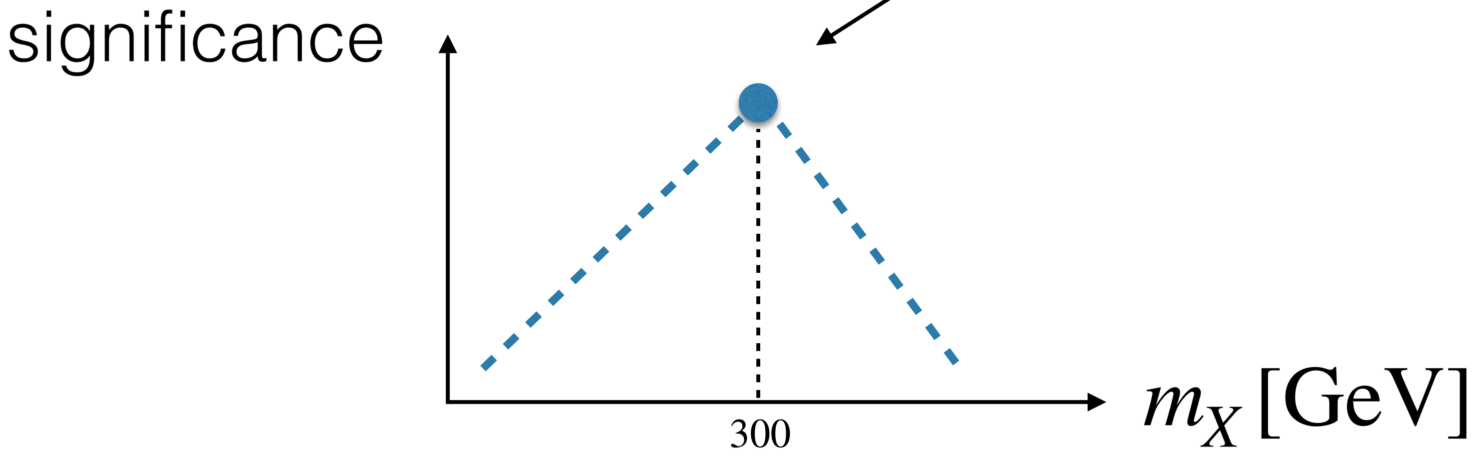
MC for $m_X = 300\,\mathrm{GeV}$

feed MC samples
to the army of Monkeys

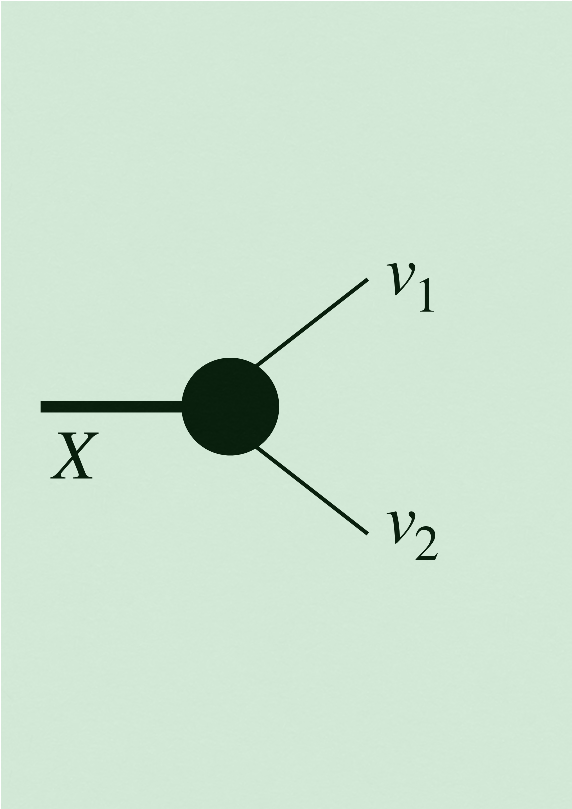


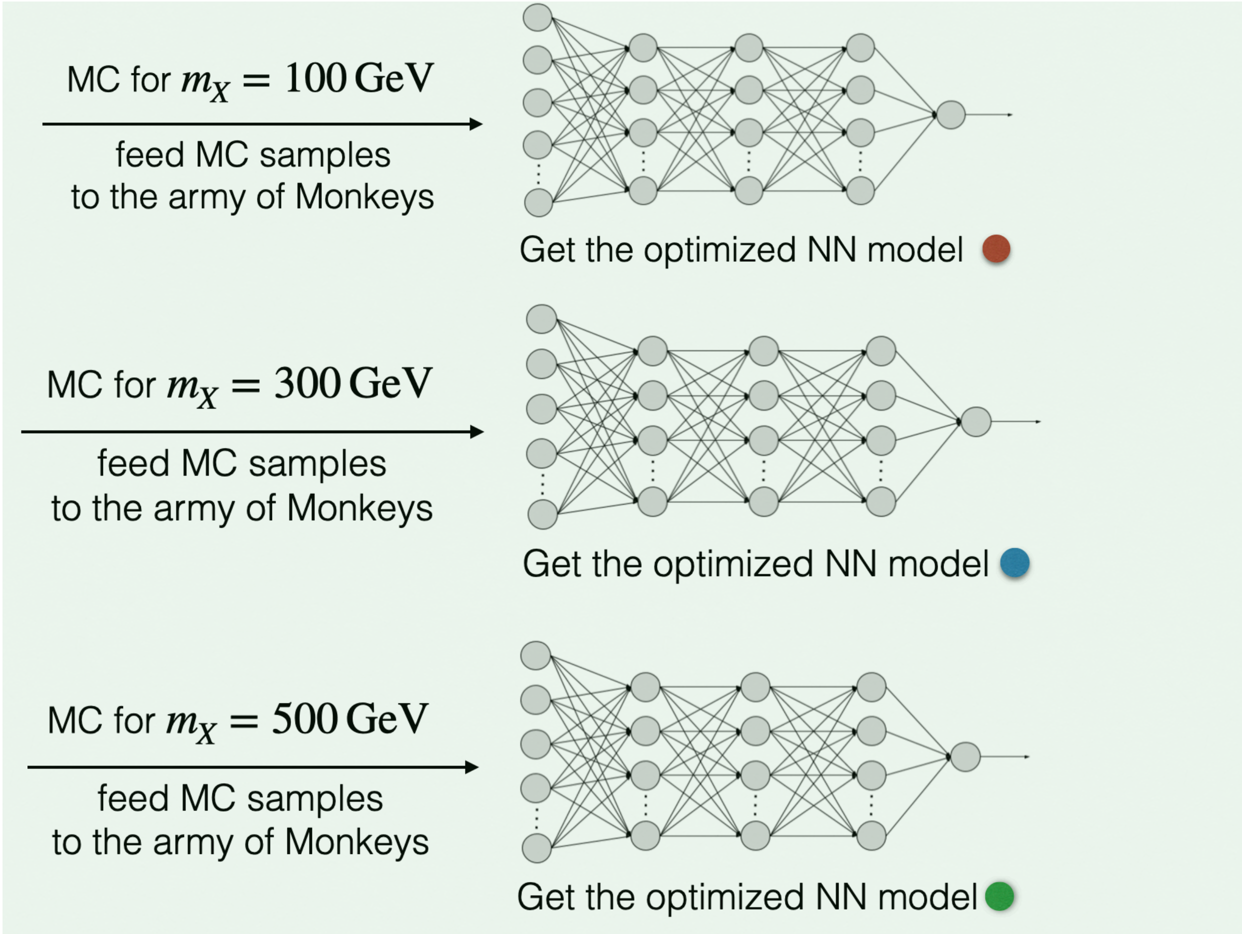
Get the optimized NN model

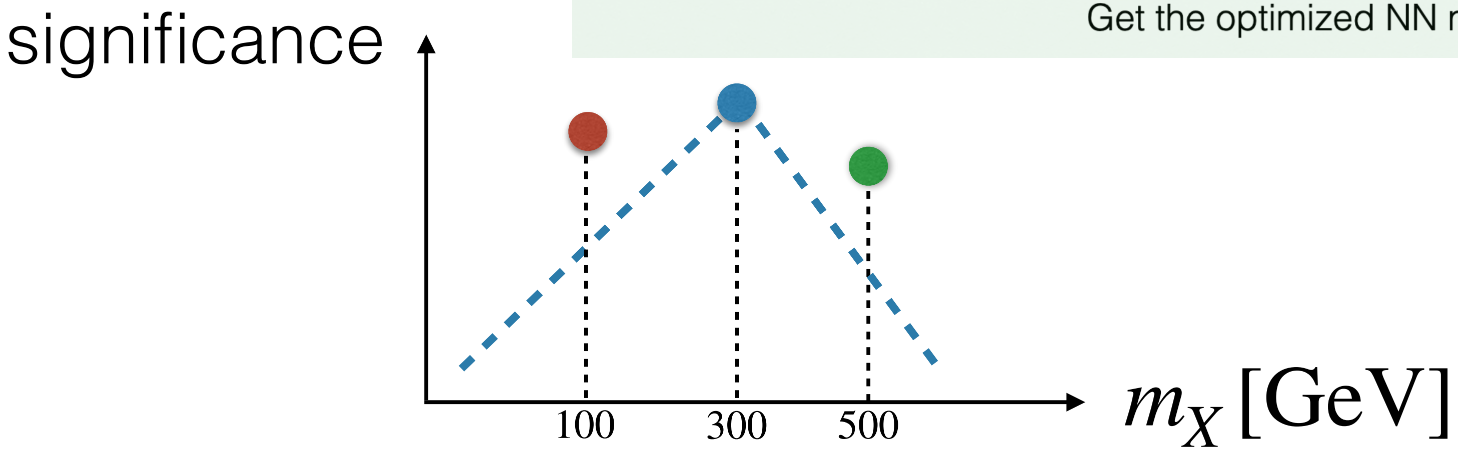
(= Save parameters of NN)



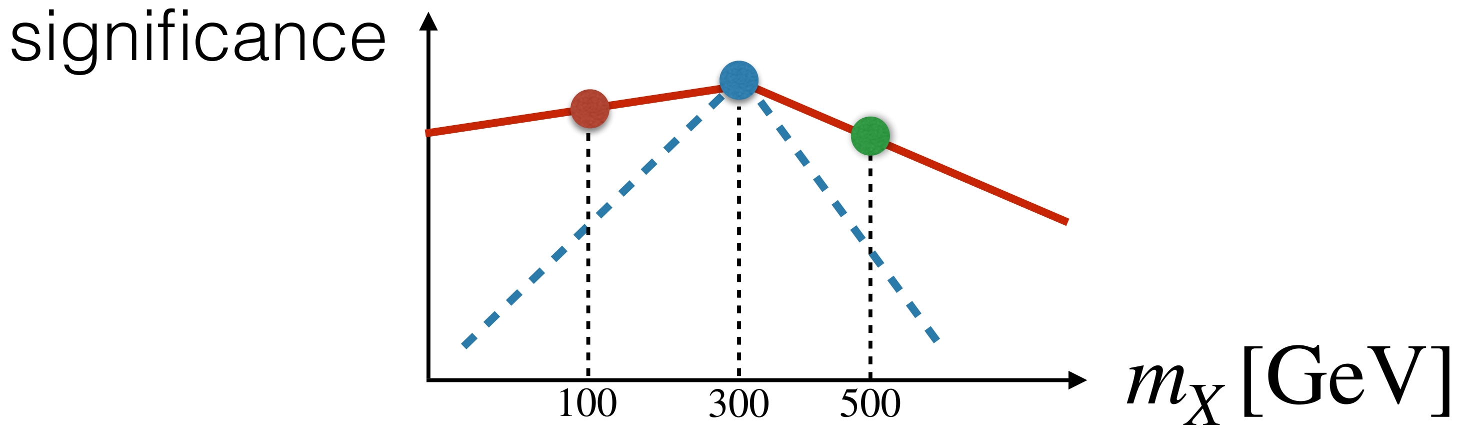
- Conventional supervised ML,



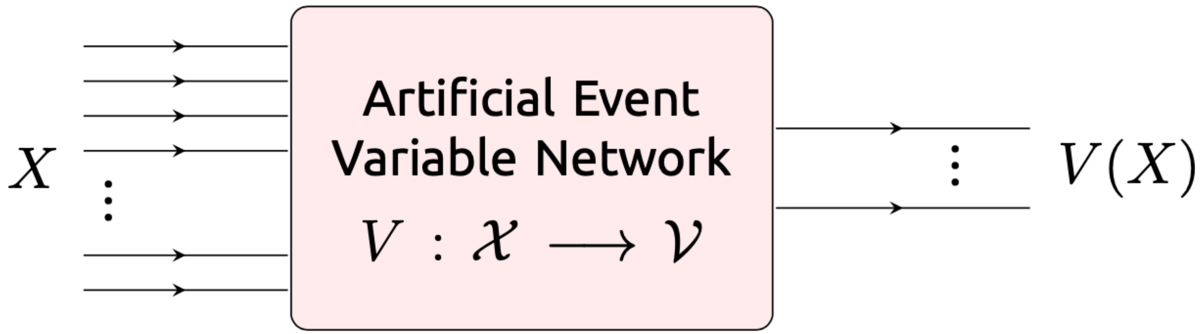




- Our **NN method** provide

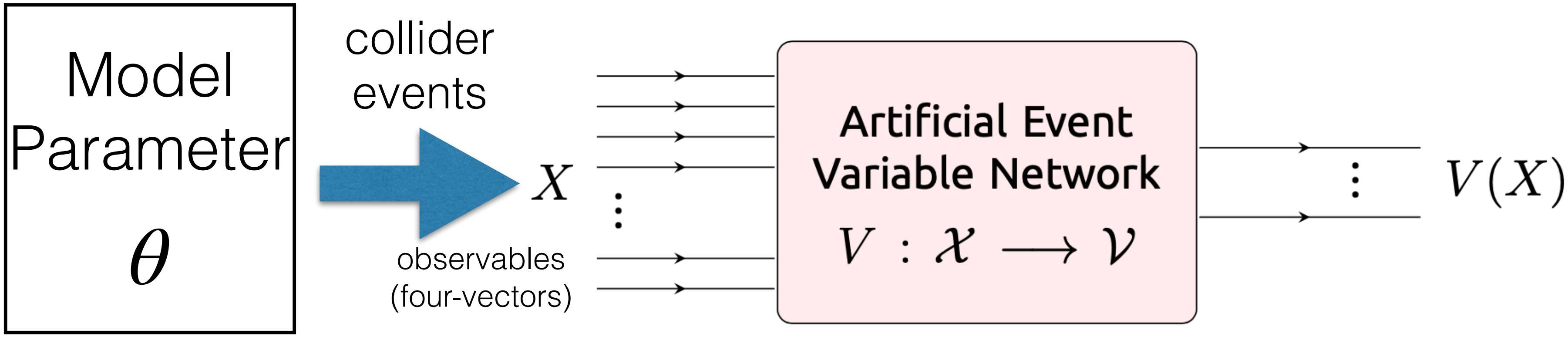


by synthesizing event variable with ML



- V **is not signal vs background classifier**

- We train the network so that V carries information about the underlying unknown parameter θ



dim(X) —— Dimensional reduction ——→ dim(V)

- Some information would be lost due to $\dim(X) > \dim(V)$
  - Try to minimize the information loss
  - Efficiently retain the underlying parameters θ

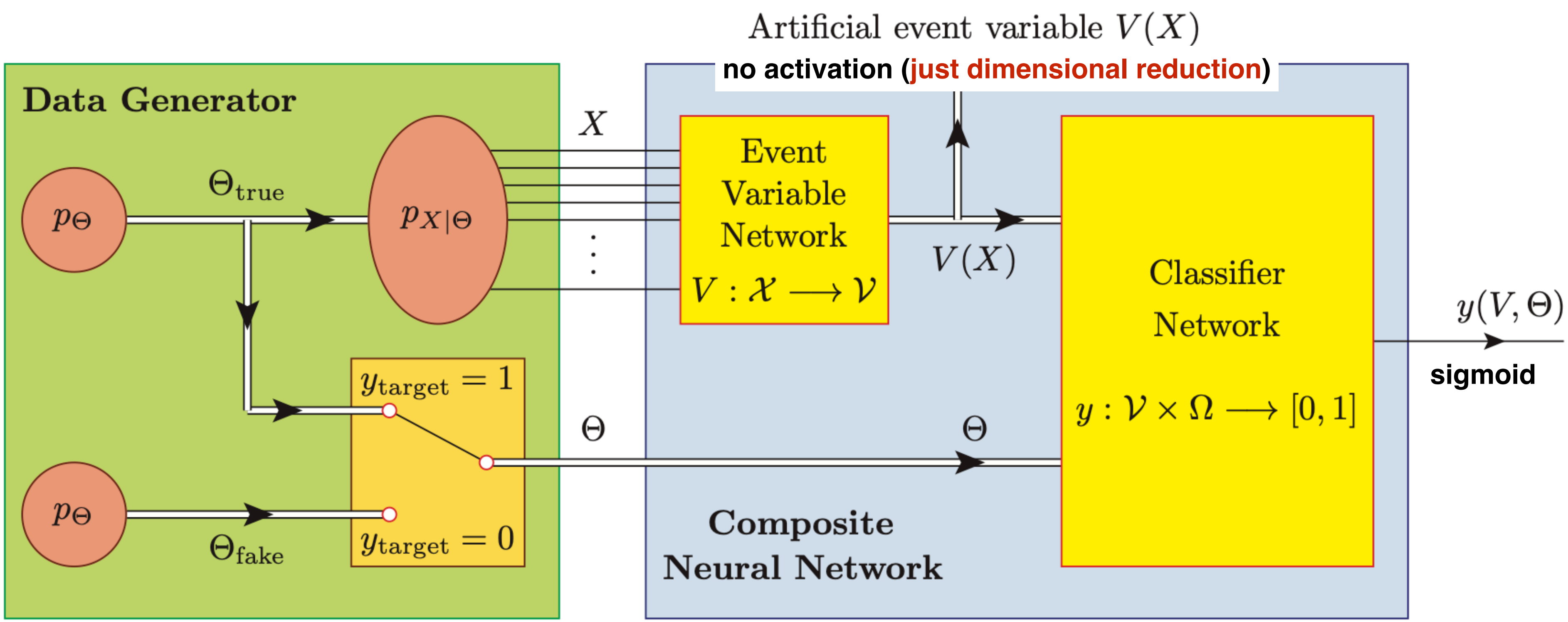
$$I(V;\Theta) = \int dv \int d\theta \, p_{V,\Theta}(v,\theta) \, \ln\left[\frac{p_{V,\Theta}(v,\theta)}{p_V(v)\, p_\Theta(\theta)}\right]$$

- This is KL divergence between $p_{(V,\theta)}$ and $p_V \otimes p_\theta$

Train V so that $p_{(V,\theta)}$ and $p_V \otimes p_\theta$ are highly distinguishable

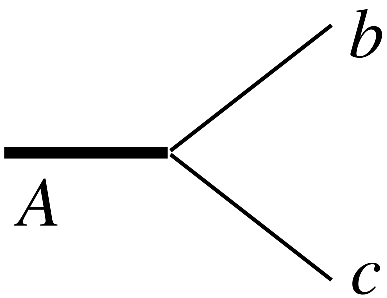
How to train V ?

- Use two classes
  1. $p_{X|\theta}$: events X **from** θ
  2. $p_X \otimes p_\theta$: events X **not from** θ (totally uncorrelated)



Does EV work?

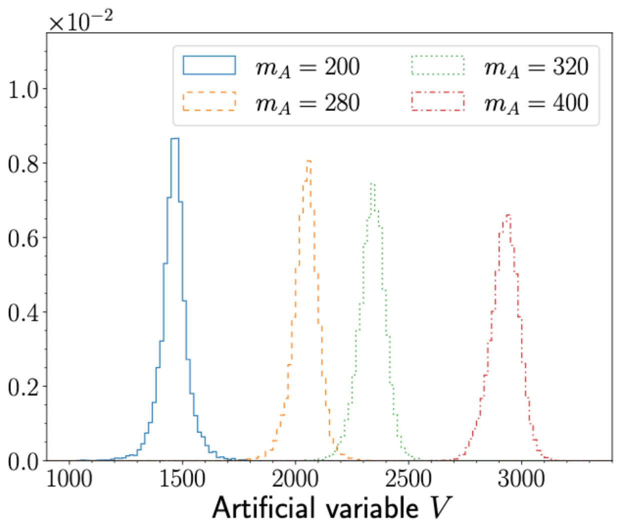
- Simplest example: Invariant mass



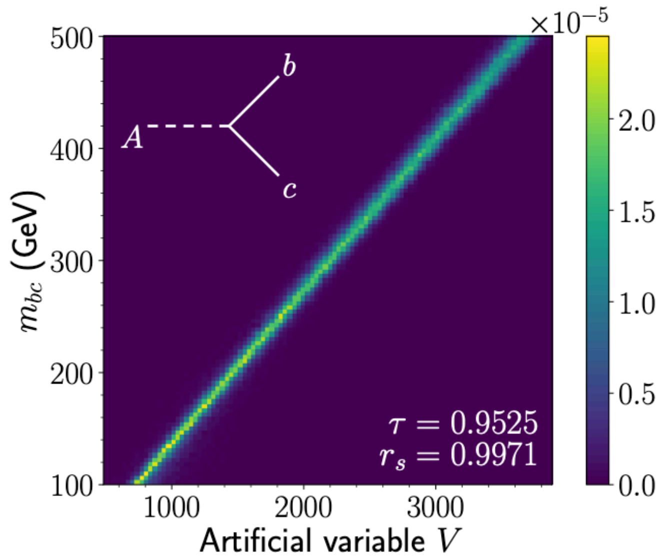
$$\theta = \{m_A\} \xrightarrow[\textbf{phase-space generator}]{} X = \{p_b^\mu, p_c^\mu\}$$

$$\dim(X) = 8 \rightarrow \dim(V) = 1$$

- Compare "Human-engineered EV =m_{bc}" and ML-EV



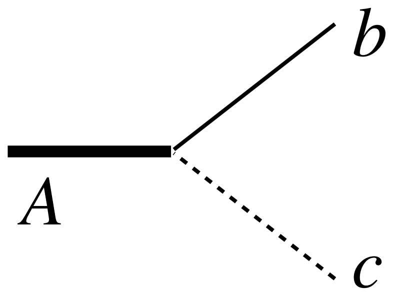
Event variable in action



What has the Machine learned?

Kendall's τ coefficient, Spearman's rank correlation r_s

- Next Simple example: Transverse mass



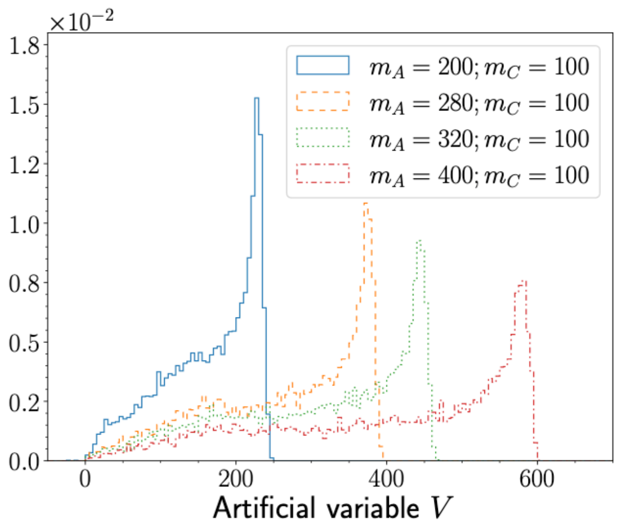
(invisible)

$$\theta = \{m_A\} \xrightarrow{\text{phase-space generator}} X = \{p_b^\mu, p_c^T\}$$

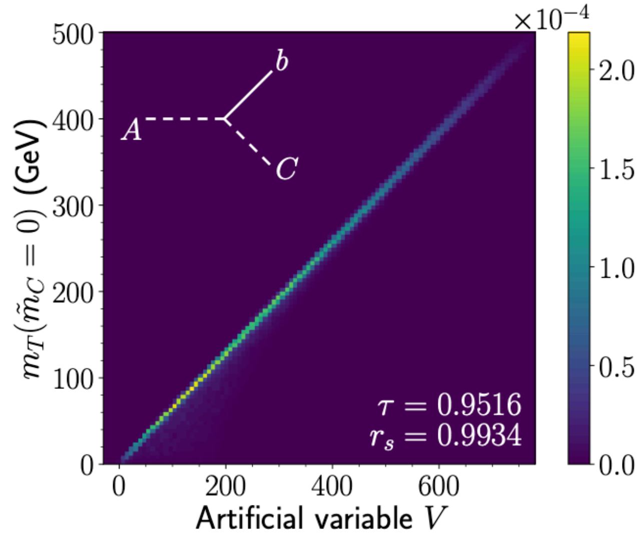
(No ISR, only beam-direction $z-$boost)

$$\dim(X) = 6 \rightarrow \dim(V) = 1$$

- Compare "Human-engineered EV =m_T" and ML-EV



Event variable in action



What has the Machine learned?

For Kinematics

- We have developed a **ML architecture** to find a "singular" variable to **describe a given kinematics** very well.

  - From now on, No human engineered feature variables required.

- what would be the next?

Orthogonal information to the Kinematics

- **Differences in radiation patterns of a Gauge charge** are coming from **"soft P_T" region :**Telling us about the state under a gauge group

Case of the Standard Model Gauge group

- In many cases, the **soft QCD radiation patterns** from signals are different from Backgrounds. (e.g. : rapidity gap)

- More detailed calculations on color flow and QCD dipole, arXiv:2407.04897 by Andrew Larkoski.

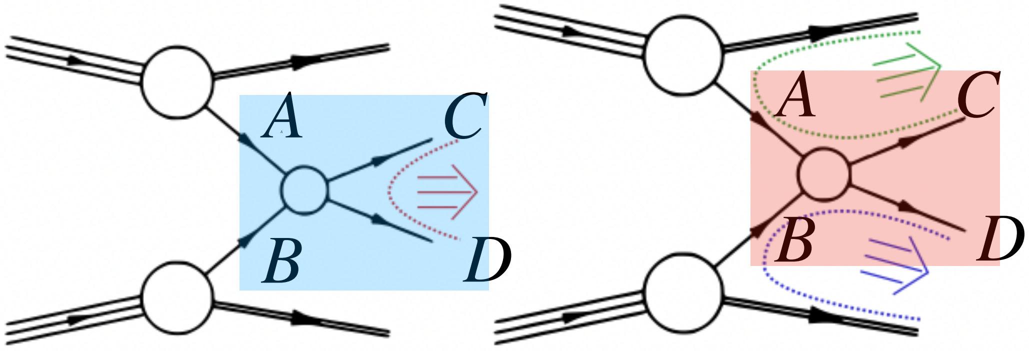
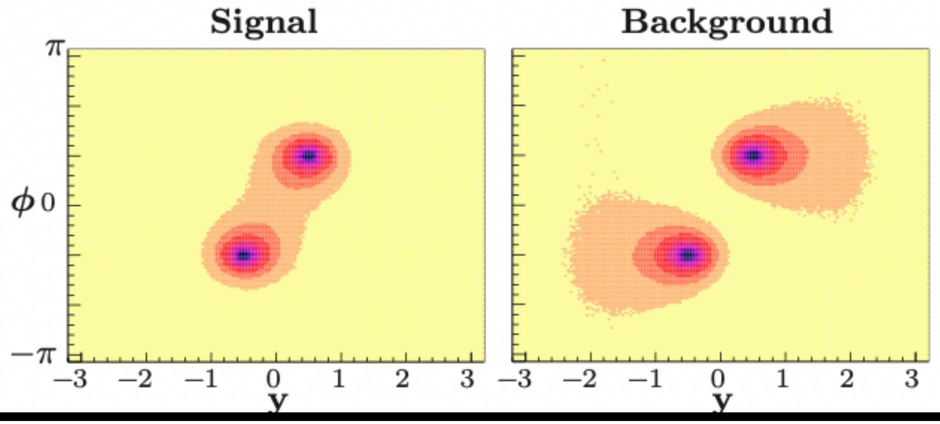


FIG. 1: Possible color connections for signal ($pp \to H \to b\bar{b}$) and for background ($pp \to g \to b\bar{b}$).



$$gg \to h \to b\bar{b}$$

$$\mathrm{Tr}[T^A T^B]\mathrm{Tr}[T^C T^D]$$

V . S .

$$gg \to b\bar{b}$$

$$\mathrm{Tr}[T^A T^C]\mathrm{Tr}[T^B T^D]$$

$$\mathrm{Tr}[T^A T^D]\mathrm{Tr}[T^B T^C]$$

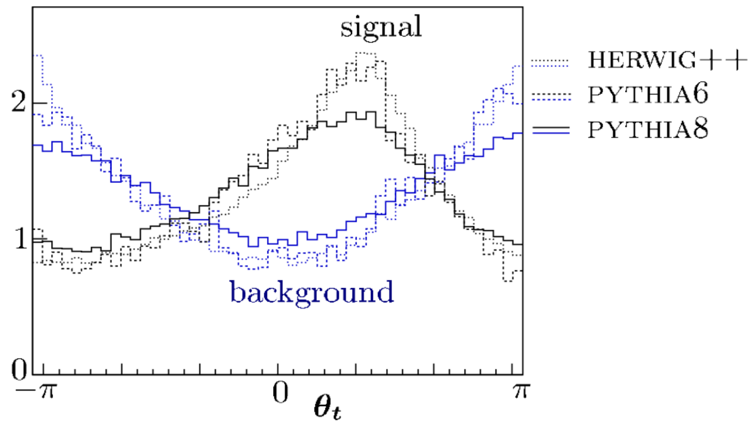
Utilizing localized information

- One can design a QCD variable, for example a pull-vector

$$\vec{t} \equiv \sum \frac{p_T^i |r_i|}{p_T^{\text{jet}}} \vec{r}_i$$

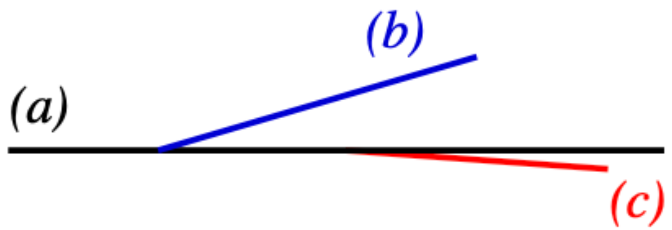
(Jason Gallicchio, Matthew D. Schwartz 2010)

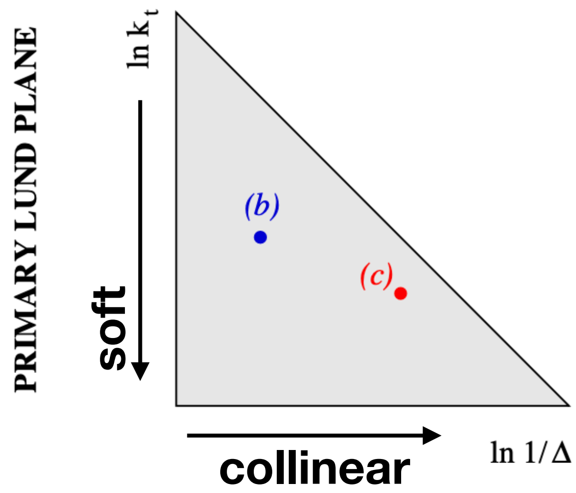
provides an one-dimensional feature



- Or one can get two-dimensional features,

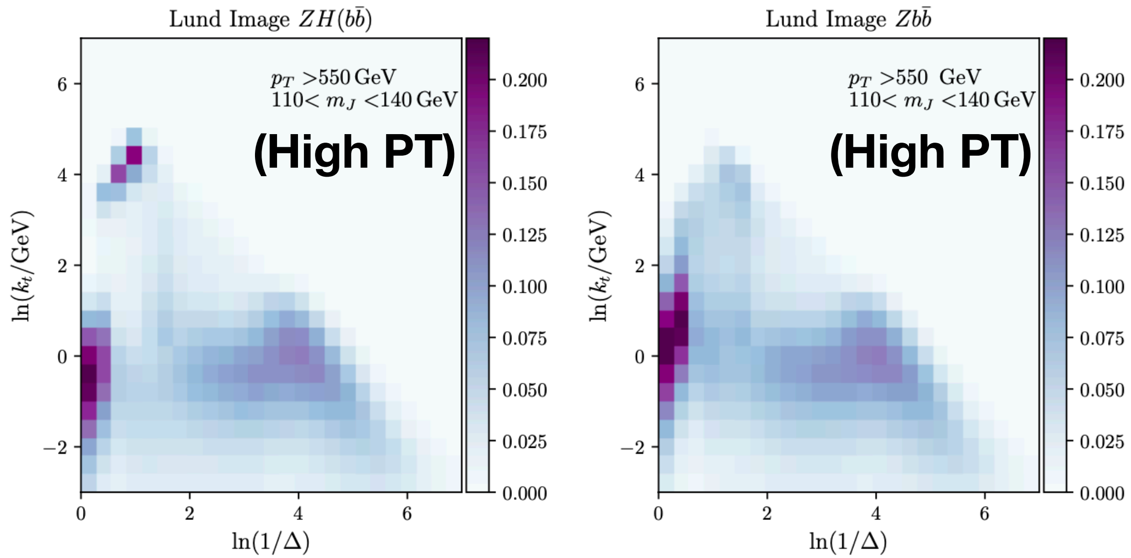
(Frederic A. Dreyer, Gavin P. Salam, Gregory Soyez 2018)



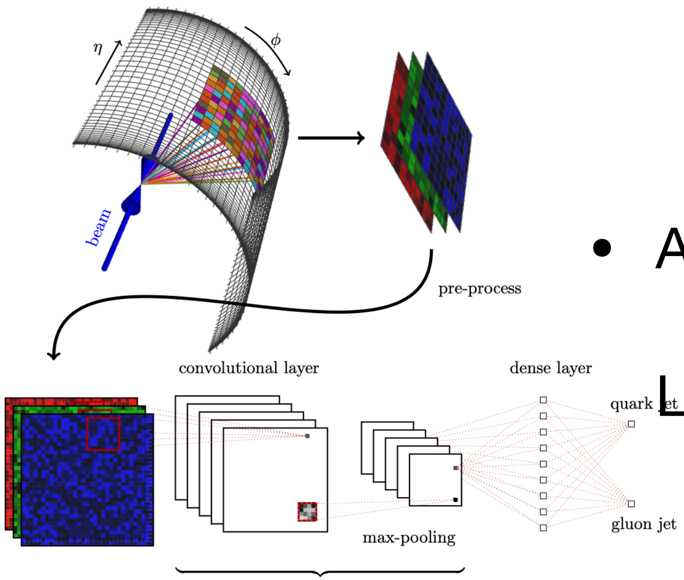


Fully utilizing localized information

- One needs to understand differences in **"the full information"**



(Charanjit K. Khosa, Simone Marzani, 2021)



M. Schwartz et.al. (JHEP 2017)

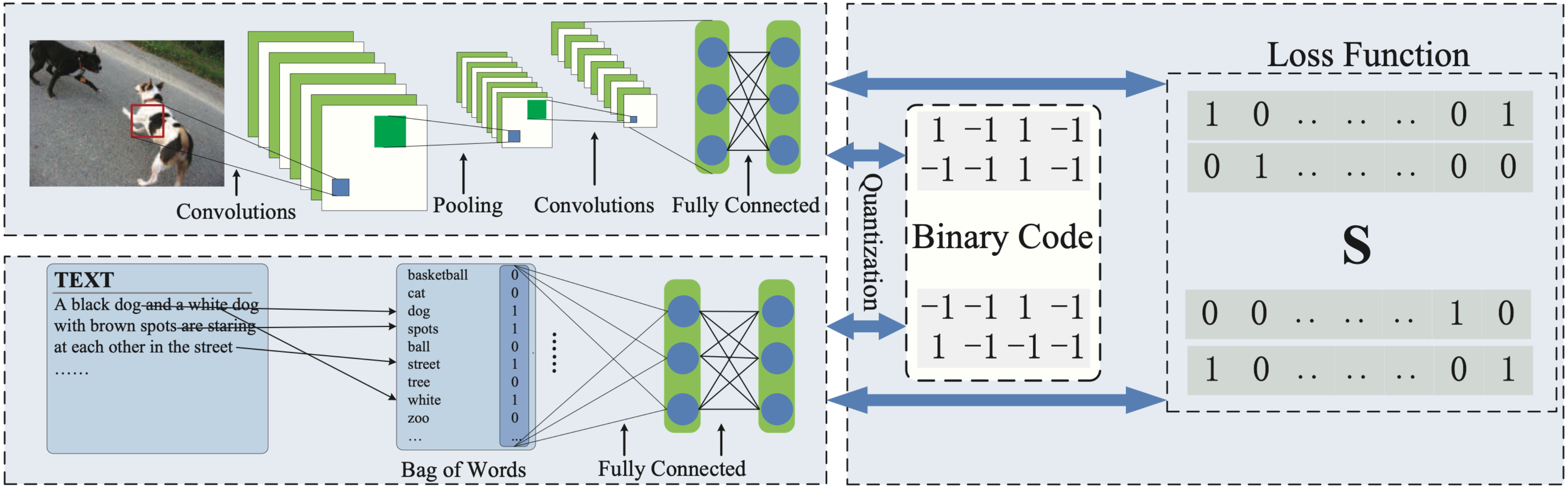
- A **neural network** can tell differences using pixels

Low level information→ feature engineering →NN

Combining two different information

?

Multi-modal Network



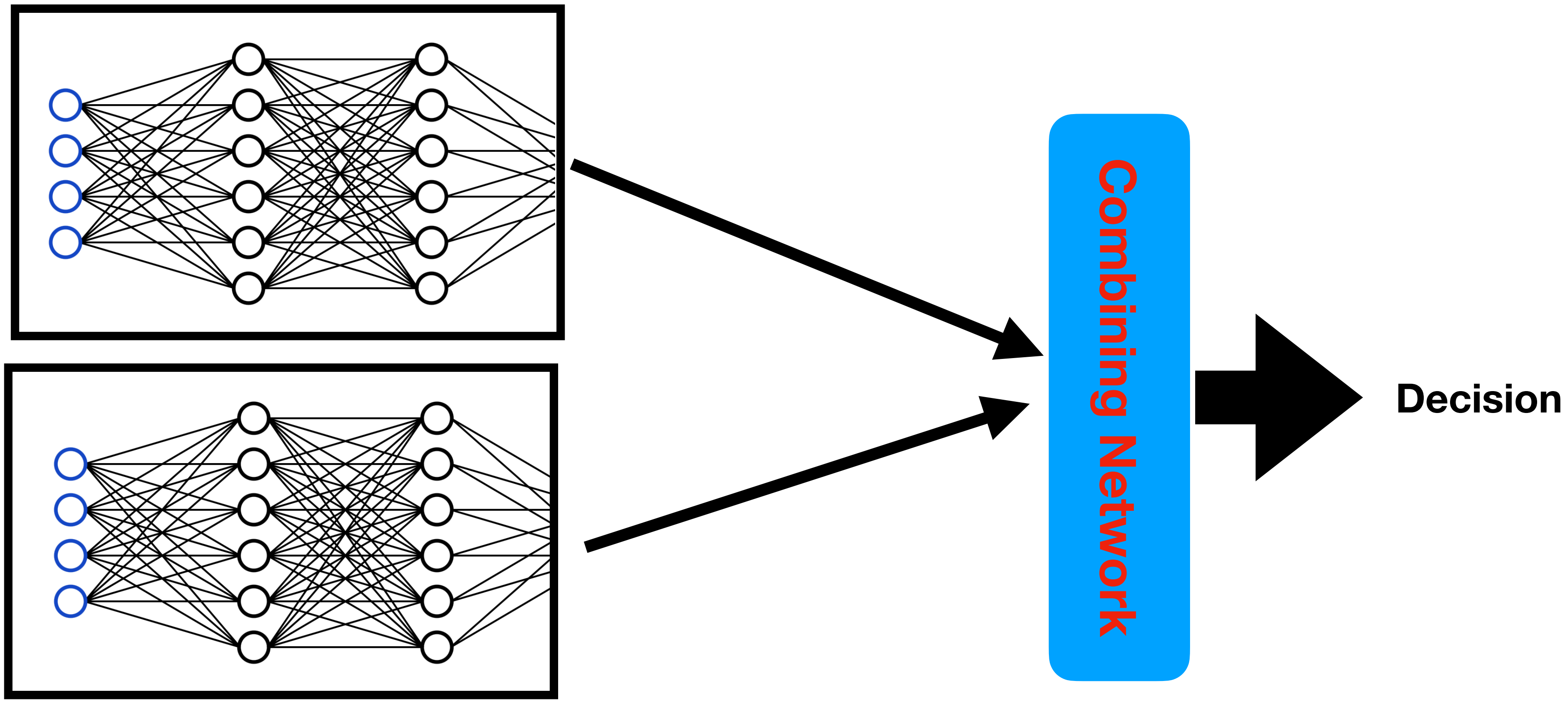
- In the commercial applications of Machine Learning, various sources of information (for example, different images, text) are utilized to interpret a situation in a consistent way.

- Here, we take the concept of Multi-modal Network
  - to extract (1) local features from QCD activities
  (2) global features from Kinematics
  - and to combine these two streamline.

Global and Local information

- Separate Neural Network during information bottleneck process
- Combination layer before decision

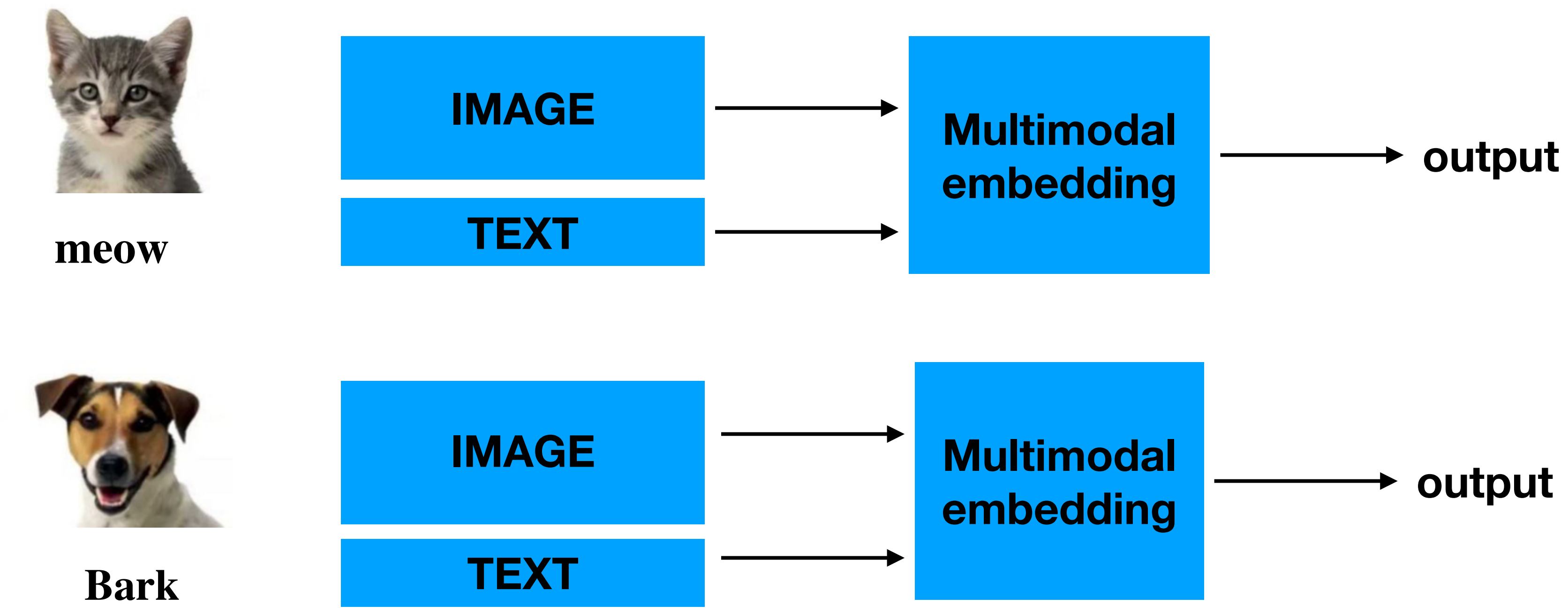
A Neural Network for "Global" information (kinematics)



A Neural Network for "localized" information (QCD activities)

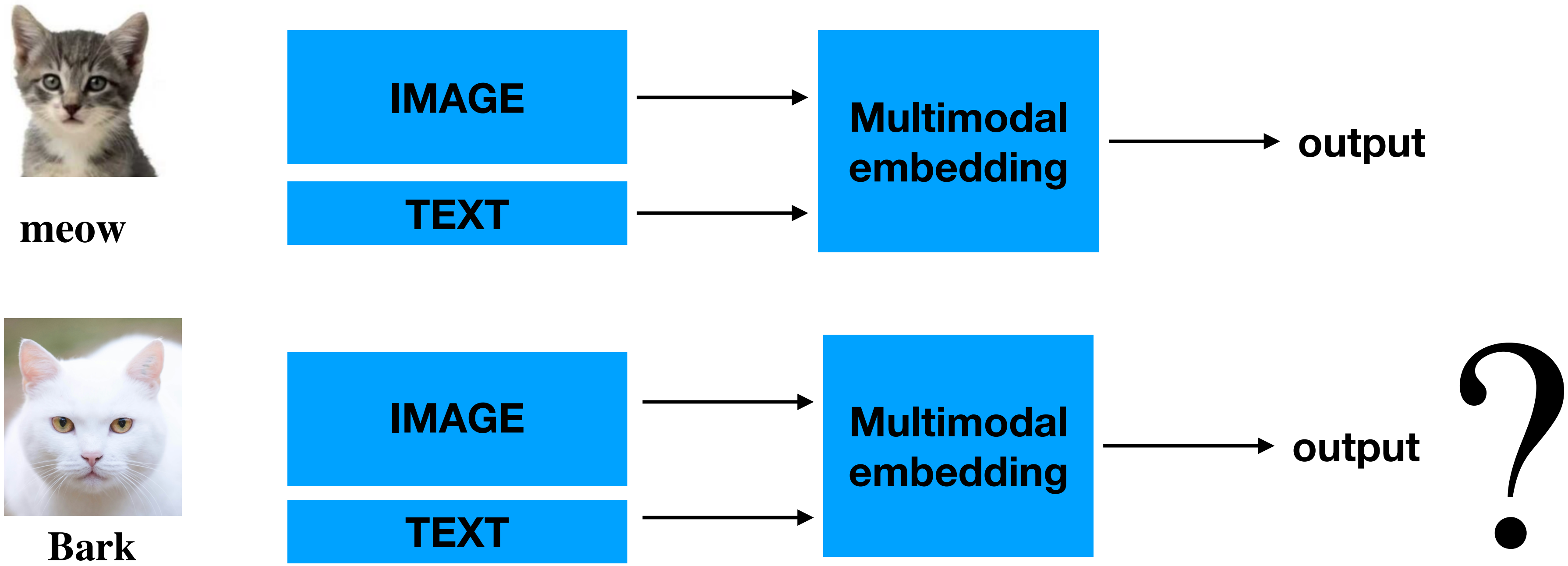
Multi-modal Network in conventional applications

- Commercial Multi-Modal application to enhancing the accuracy through **consistent** patterns across modalities

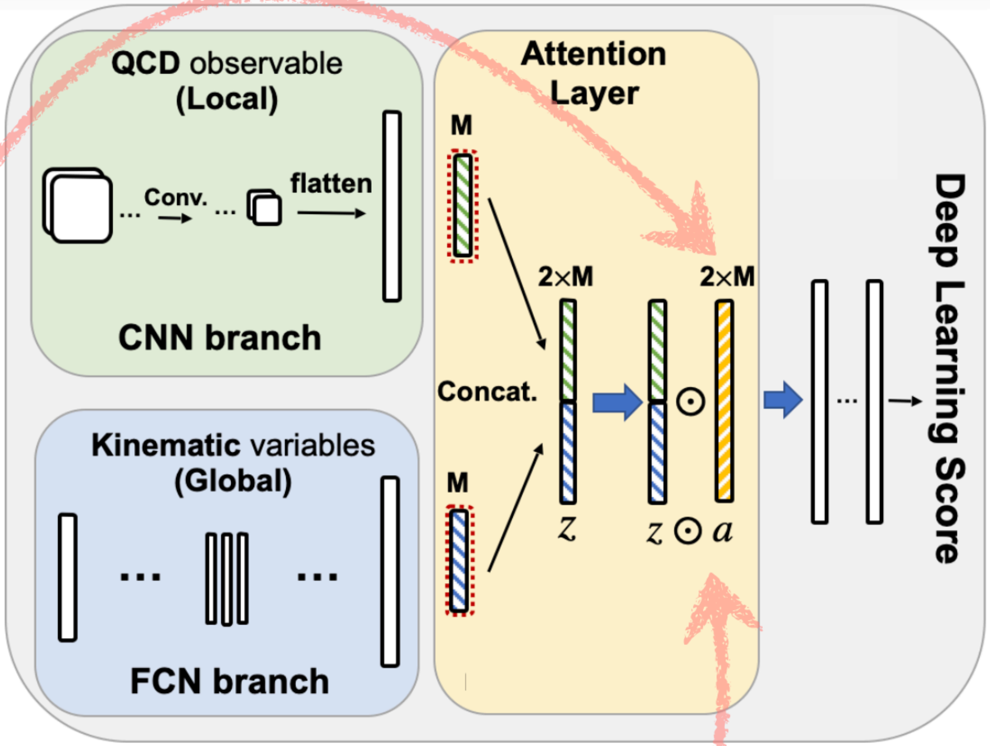


Multi-modal Network in collider physics

- In the collider physics, "kinematics" and "localized pattern" (for example QCD) are **orthogonal information**,
  so that <u>we need to take a **complementary** approach.</u>



Attention mechanism inspired by NLP



Attention score → $a(z) = \frac{e^{f_{attn}(z)}}{\sum_{j=1}^{2M} e^{f_{attn}(z)_j}}$

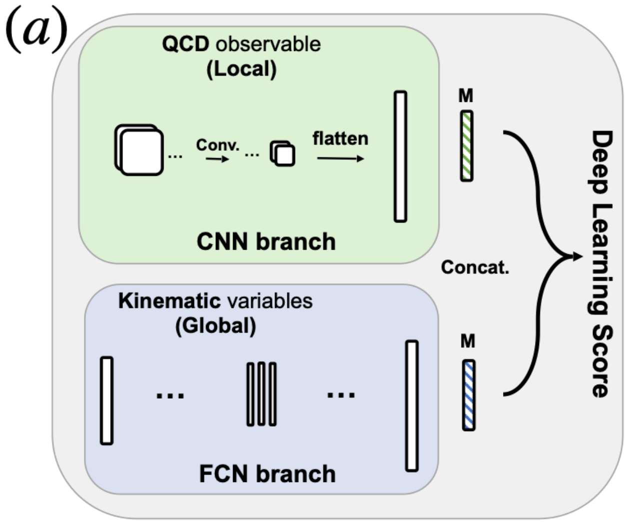
where the layer $z = [z_{1:M} : z_{M+1:2M}]$ which is the concatenated layer from CNN and FCN respectively, and $f_{attn}(x) = W_{attn}x + b_{attn}$ is a trainable linear transformation.

Attention value → $z \odot a$

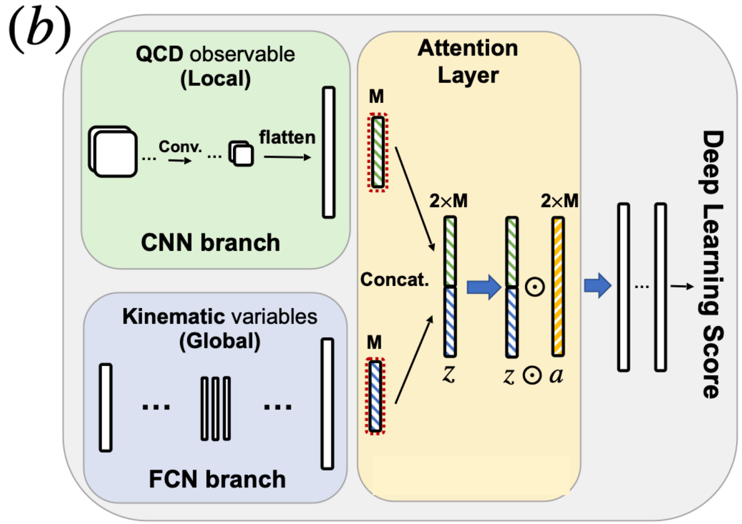
✓ The attention values are computed as Hadamard product between the attention score (a) and the concatenated layer (z).

✓ We can interpret how much the model concentrates the two base models to classify signal and background.

✓ Attention values are then connected with a fully connected layer for classification.



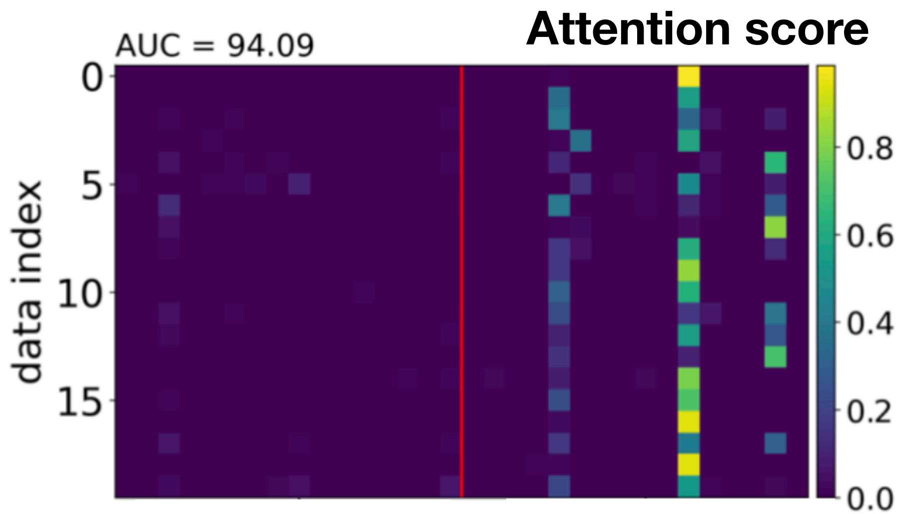
Conventional
Black-box NN



Transparent NN

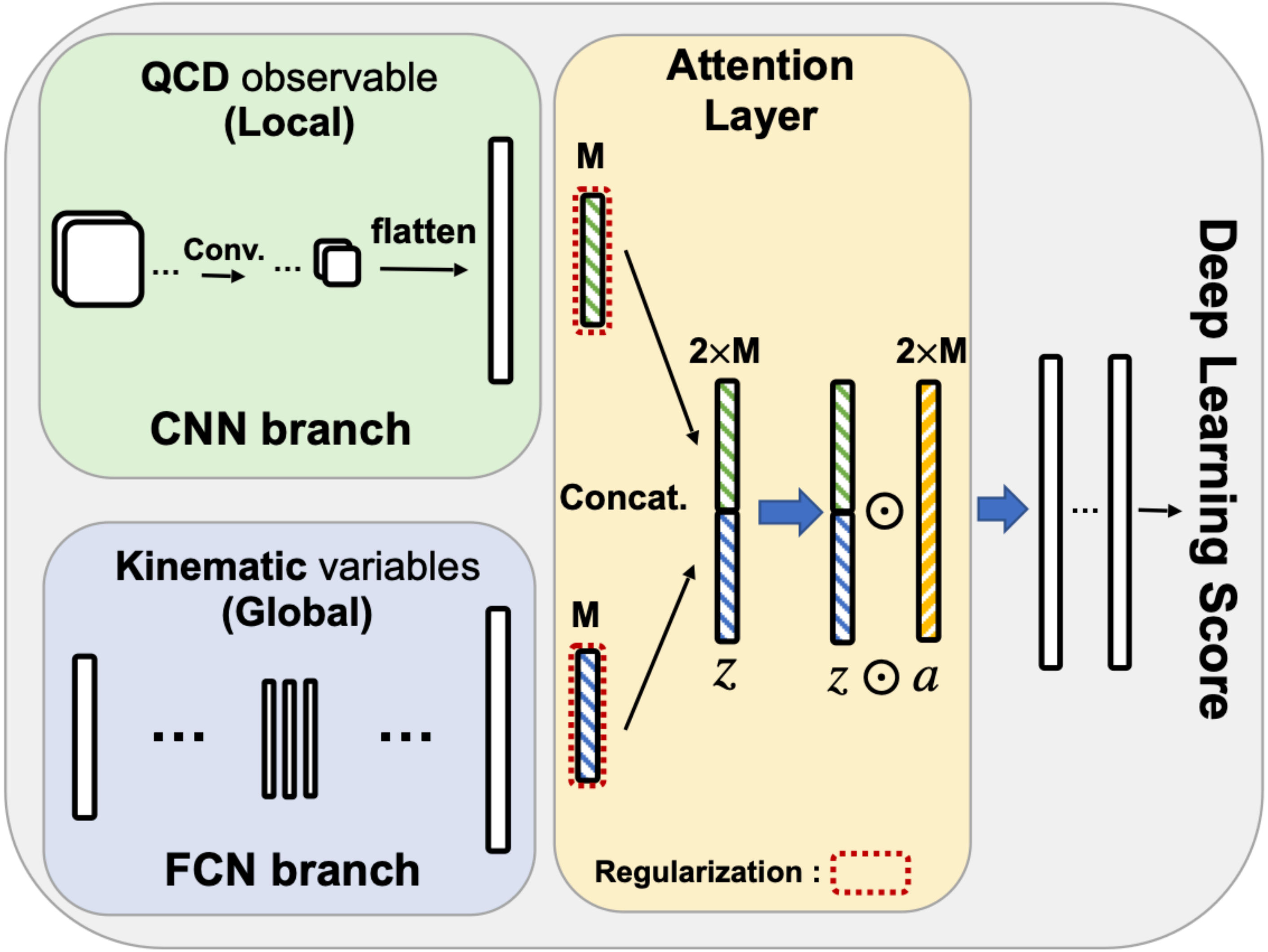
- With **an attention layer**, <u>we can observe</u> that

  Network with the conventional combination **focus mostly on the easier material to study. (kinematics)**



Network with QCD soft patterns

Network with Kinematics

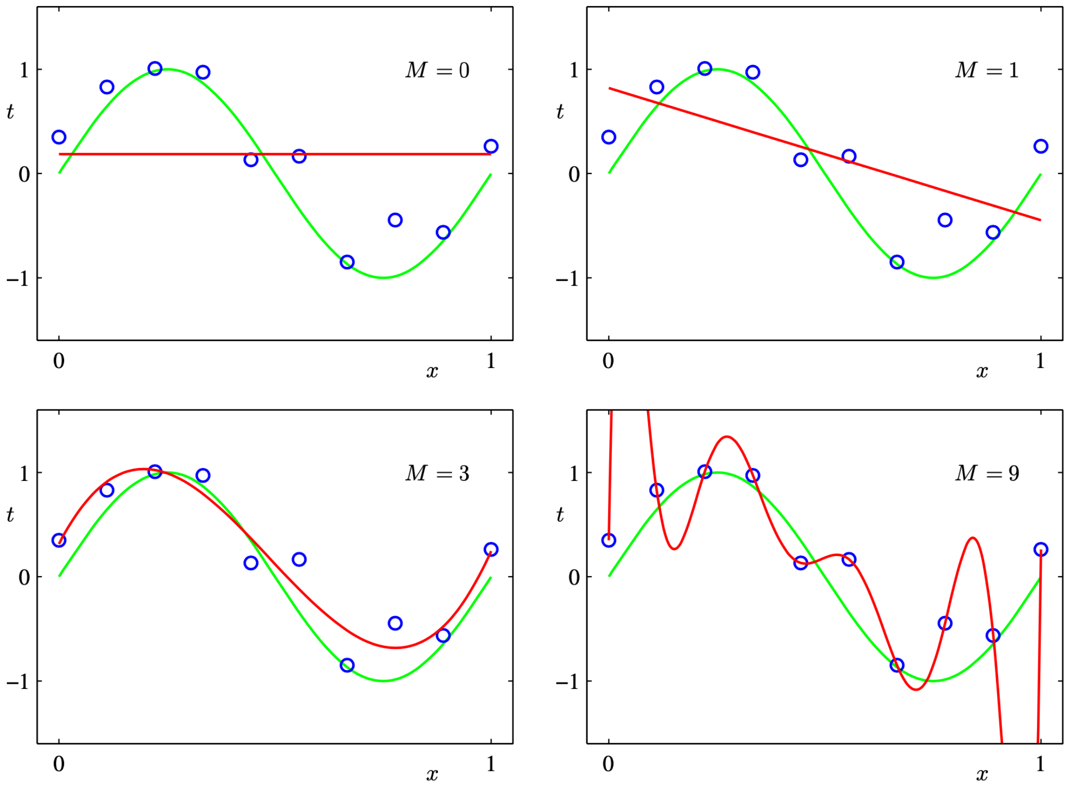


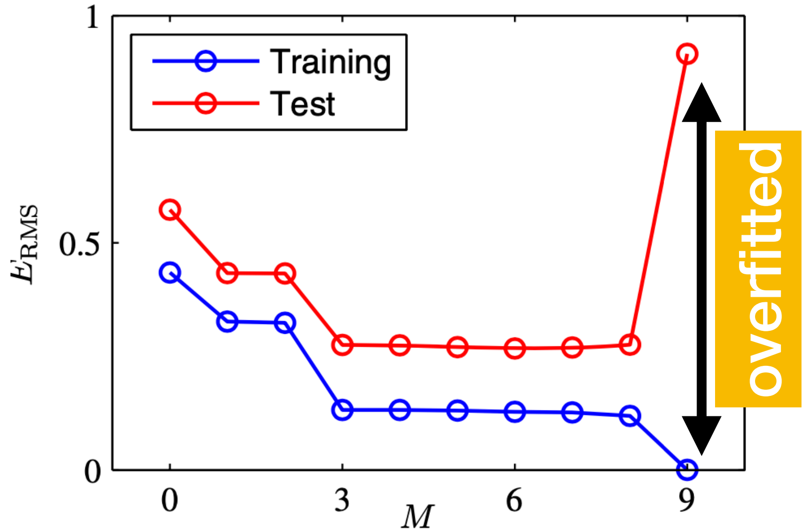
$$\text{Regularization term} = l_2 \times \sum_{k=1}^{M} W_k^2 \quad \mathcal{O}(10^{-2})$$

Actually, this is related to the core of ML - protecting overfitting

- With complicated networks, the overfitting can occur !

$$y(x, \omega) = \omega_0 + \omega_1 x + \omega_2 x^2 + \cdots + \omega_M x^M$$



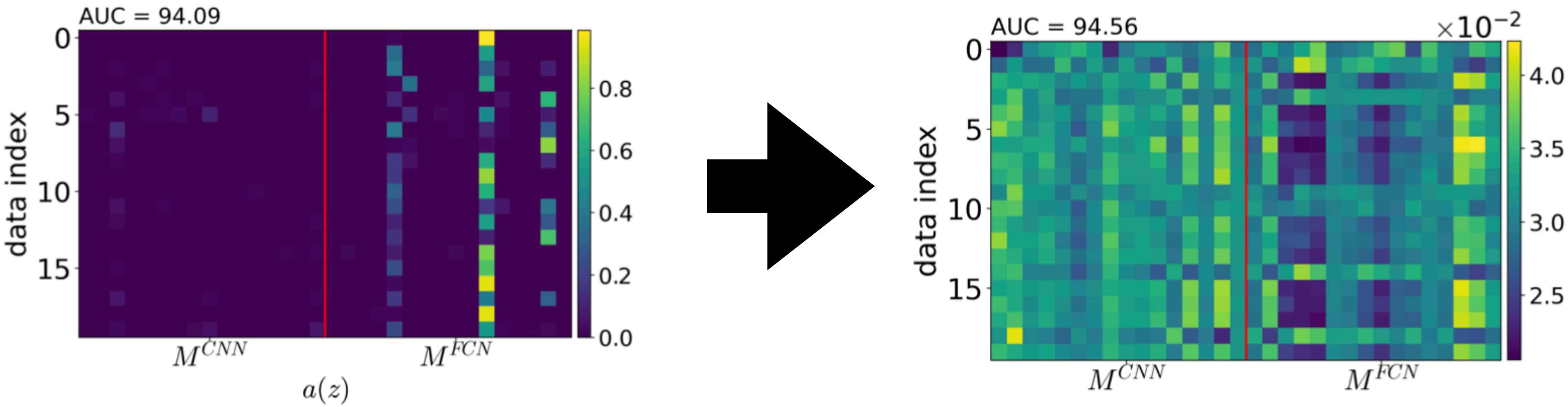


	$M = 0$	$M = 1$	$M = 6$	$M = 9$
$w_0^\star$	0.19	0.82	0.31	0.35
$w_1^\star$		-1.27	7.99	232.37
$w_2^\star$			-25.43	-5321.83
$w_3^\star$			17.37	48568.31
$w_4^\star$				-231639.30
$w_5^\star$				640042.26
$w_6^\star$				-1061800.52
$w_7^\star$				1042400.18
$w_8^\star$				-557682.99
$w_9^\star$				125201.43

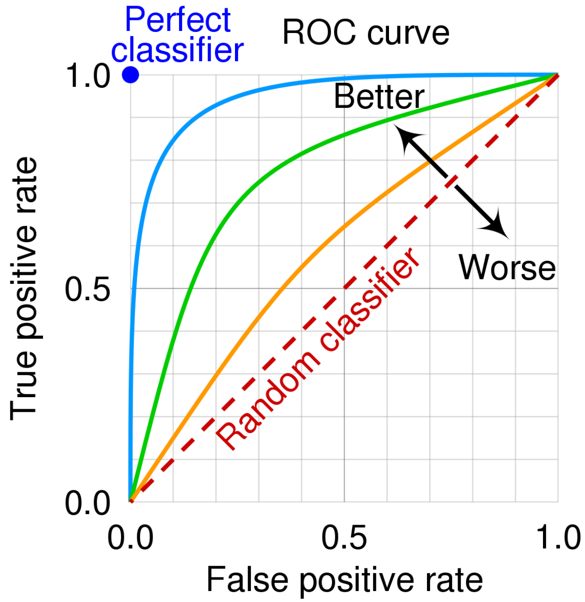
- Utilize a **regularizer**, a"**Lagrangian Multiplier**" in ML

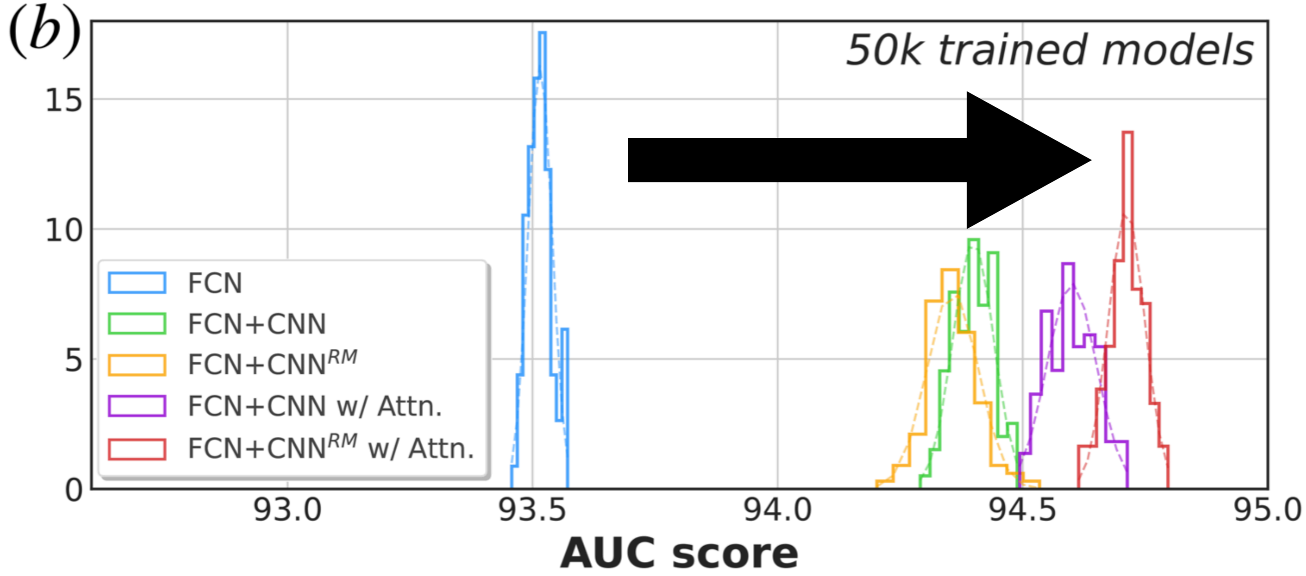
$$\text{Regularization term} = l_2 \times \sum_{k=1}^{M} W_k^2$$

- this **simple technique** works to **balance** information between local and global information!



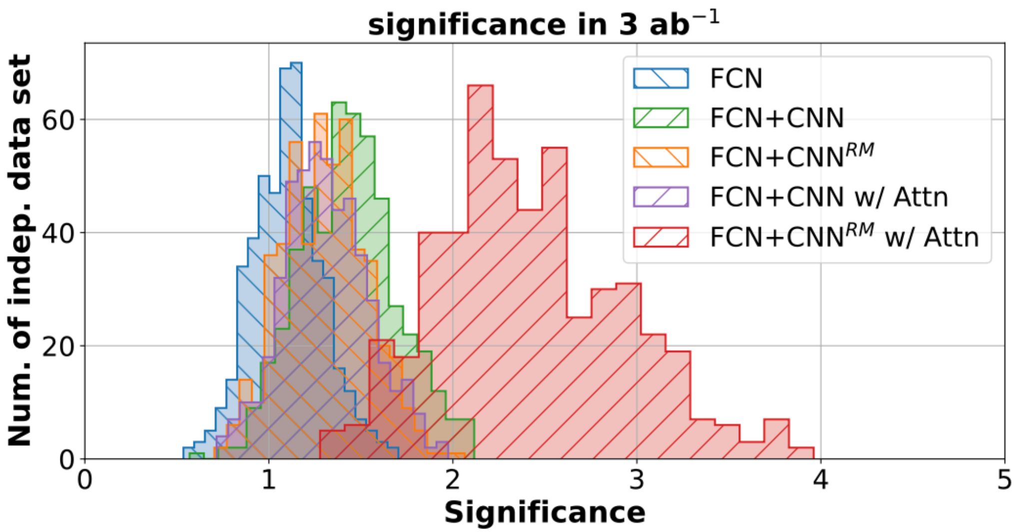
- A simple but effective solution.





- AUC score is the area under the ROC curve.
  Perfect discrimination = AUC score=100

- We get the improvement in the performance as a result.



AI for Collider Physics

- **Efficient parameter exploration**
  - Precision measurements impose increasingly complex constraints on theory parameter spaces.
  - AI helps us efficiently explore the surviving regions consistent with experimental data.

- **Efficient Monte Carlo simulation**
  - As collider measurements become more precise, theoretical predictions must keep pace.
  - AI-assisted Monte Carlo techniques improve the precision and efficiency of theoretical calculations.

- **Enhanced new-physics discovery**
  - New physics signals can be subtle and distributed across multiple observables.
  - AI can combine complementary information more effectively, improving the sensitivity of collider searches.